(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 February 2005 (17.02.2005)

PCT

(10) International Publication Number
WO 2005/015404 A2

(51) International Patent Classification[7]: G06F 11/30

(21) International Application Number: PCT/GB2004/003383

(22) International Filing Date: 6 August 2004 (06.08.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/492,842 6 August 2003 (06.08.2003) US

(71) Applicant and
(72) Inventor: HALEVY, Moshe [IL/GB]; 33 Malvern Road, Hampton, Middlesex TW12 2LW (GB).

(74) Agent: HILL, Justin, John; McDermott, Will & Emery, 7 Bishopsgate, London EC2N 3AR (GB).

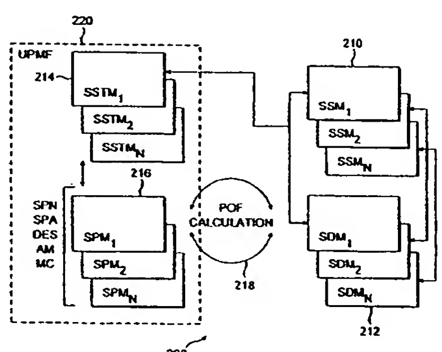
(81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND APPARATUS FOR UNIFIED PERFORMANCE MODELING WITH MONITORING AND ANALYSIS OF COMPLEX SYSTEMS



(57) Abstract: A method and apparatus (200) for unified performance modeling with monitoring and analysis of complex computer systems is disclosed. An aspect of the invention relates to performance monitoring and analysis by constructing a synthetic work load of the complex system and monitoring a parameter of the system indicative of system performance. A system deployment model and system scenario model pair are constructed from worst case scenarios of the parameter over a period of time. A system performance model (216) is extracted from system state model (214) associated with the system scenario model (210). The system performance model is modeled for point of failure occurring in the system. Another aspect provides a unified performance modeling to extract the system performance model from system state model (214). Additionally disclosed is an interactive system performance modeling using system scenario model and system deployment model pair with updated systems point of failure. Another aspect is a method for performance monitoring and analysis of a system in a grid computing environment.

WO 2005/015404 A2

METHOD AND APPARATUS FOR UNIFIED PERFORMANCE MODELING WITH MONITORING AND ANALYSIS OF COMPLEX SYSTEMS

FIELD OF THE INVENTION

The invention generally relates to performance analysis, for example, targeted at enterprises or third party service providers worldwide. More particularly, the disclosed embodiment relates to a method and system that facilitates information technology (IT) and telecommunication performance analysis services for most enterprises worldwide at an affordable price.

BACKGROUND

System Performance is often the central aspect during a software lifecycle of design, development, and configuration–tuning. It is not always enough to know that systems work properly, they must also work effectively. Research in several areas such as information technology (IT) and telecommunication, military, health care, financial-services, etc, has shown that resources of time, money, (lives!), can be saved if the performance of a system is optimized. Performance analysis studies are conducted in order to evaluate existing or planned systems and to compare alternative configurations in the aim of finding an optimal configuration for a system. Currently Performance Engineering, is split into three categories: Performance Monitoring (Measurement), Load-Testing (Benchmarking) and Performance Modeling (Simulation, Analytical and Numerical Methods). Systems' Architecture is fixed configuration based such as client-server or dynamic configuration based such distributed system (e.g. Web Services based service oriented architecture (SOA), Agent based systems).

Back-of the-envelope performance predictions, such as raw basic machine characteristics analysis, is a very risky and ineffective method, given the complexity of performance-

engineering analysis needed in a multidimensional systems performance space. Current market solutions mainly promote performance monitoring activity, where performance modeling is provided as a supplement.

With respect to enterprise's systems cost reduction measures, as much as one would like to take a structured approach towards cost cutting, a balance must be kept so a system does not fail to support the business-operations. Precise and cost-effective methods in managing these challenges are needed, but it is a reality that decisions concerning systems performance are based on guesswork by most companies. In other cases there is a degree of limited usage of performance engineering practices such as load testing, performance monitoring of transactional or web application and software-vendors performance-optimization (tuning) applications (e.g. such as applications from SAP of Walldorf, Germany, and Oracle of Redwood Shores, California, United States of America).

In current operation/application managers who require delivering the expected subject-matter system' service levels are doing their job "blind-folded". This is mainly as the means available by current art to conduct a proper performance engineering analysis are limited due to:

1) Mounting costs. If one tries to integrate various existing performance engineering tools into one, its costs and particularly their operator labor costs will accrue to a large sum; hence not COST EFFECTIVE but to the very large corporations;

2) Such tools are NOT well integrated;

3) Seamless and complete analysis enabling a 'big picture' analysis of how enterprise's systems architecture is utilized is by concept and technology is not available in the market.

4) Performance analysis based on load testing is inflexible, since the results only relate to a specific architecture at a fixed point in time. Hence model based load testing became obsolete with relatively small architecture's design variation.

5) Performance modeling tools based statistical-analysis and mathematical-modeling techniques are very expensive, require major in-house resources to operate, and typically only focus on a particular aspect of the overall system.

6) Performance monitoring solutions typically report problems too early, resulting in unnecessary expenditure, or too late, resulting in system failure.

This magnitude of such lack of knowledge is increasingly becoming a major factor of concern for many businesses as 'systems' performance is degraded in direct correlation to the increased complexity derived from today's system's architecture complexity.

The bottom-line impact of the current overall limited performance analysis practices is VERY significant: 'overkill' hardware (e.g. low capacity utilization average of 60% or less, based on survey of Chief Information Officers (CIO) conducted by Mark Techstrat of Merrill Lynch the investment bank, published on March 11th 2002), business operations interruption as a results of 'systems' performance failure that is based on on-going or change-request operations, mounting re-development costs in new systems development/integration projects where system's architecture re-design is needed as performance is well below requirements. In dollar terms a lot of waste investments and resources take place.

Fundamentally, in many companies the operational director/manager cannot tell with any confidence that: (1) He/She has cost cut as much as possible without exposing the company to the risk of losing revenue as direct result of system failure (2) That the operational function has the fundamental system attributes associated with the ability to cope with growing business demands or business dynamics. This lack of confidence originates from

lack of essential understanding of how enterprise systems (hardware and software) are utilized in relation to the business process they support.

The only service concept available is of software vendors providing generic products to Clients, with the exception of very limited performance monitoring products of Keynote Systems Inc. of San Mateo, California, United States of America, and BMC Software, Inc. of Houston, Texas, United States of America, where it is assumed that enterprise clients have the skilled resources for operating such tools. It can be argued that such concept can only work with very large organizations able to invest in retained skill Performance-Analyst staff, whose skills cover: system-metrics analysis, capacity planning, statistical analysis, mathematical modeling and system requirements skills.

Another caveat of providing software products directly to enterprises is that each product covers limited scenarios out of the overall scenarios of performance monitoring and capacity planning (by performance modeling). Hence, companies are required to purchase several software products which aren't well integrated and need expert skill to operate. This can result in relatively high costs compared with value return. Based on return on investment (ROI) analysis companies would be put off from making such an investment towards performance analysis service catering for their systems. Resulting in the mid to large size companies market not being supported by cost effective and comprehensive performance engineering.

Performance analysis involves with system's data in digital format. This data is important for system maintenance but isn't actively used in clients' production systems for business application support. It means that performance analysis services can be purchased on flexible terms from able and specialized service providers. These services can be sold and managed as commodities.

Another aspect for consideration is that the computing requirements of monitoring and modeling algorithms at large aren't feasible to process at source i.e. at clients' site. Therefore, any performance-analysis at source is limited in scope. This leads to a degrading of the value of the analysis. Scope is limited as computing performance-analysis algorithms at source will result in counter productivity rather then performance improvement. Hence, client-site analysis techniques have drawbacks.

Prior methods of providing performance analysis services do not allow separate companies to collaborate together in providing such services for enterprises seamlessly and on full outsourcing terms from anywhere. Hence, performance analysis services cannot be commoditized. It does not allow providing performance analysis as commodity for the masses, compared to a luxury service for the a few.

Current modeling-processes are tied with modeling-implementation. For example, the simulation modeling tool is separate from the analytical modeling tool. One cannot model first and then proceed to simulation or analytical or both (for calibration the method of choice) all from one interface. Hence, modeling fine-tuning interactively is over complicated. Also, one cannot extend any modeling methods within one interface, rather there is a need to use another modeling tool. The current modeling solutions cannot be scaled-out.

Modeling and Monitoring processes and data currently are loosely coupled. One can do modeling or monitoring, but cannot do both within one component's interface. This raises the following problems:

1) One cannot conduct performance monitoring (with drill down capability) based on the precise 'system architecture' model. Currently building a model from scratch is

needed while setting monitoring definitions at the monitoring component, rather than passing such a model as an object after building it in a modeling component.

2) It is not straight forward to calibrate a 'system' model by reflecting on the model image the system resources share and its forecast, all generated from monitoring objects interfaces. Hence, model building effort is increased with a negative impact on cost and time.

3) One cannot have 'big-picture' capability to view concurrently 'system's' model reflected with its monitoring data. One cannot view how enterprise systems are utilized in relation to the business process they support. Such view is indispensable as an analytical tool when dealing with performance issues or when performing a what-if analysis by modeling.

Thus, no known technology provides a method and a apparatus for performance monitoring and analysis that addresses some of the problems associated with the prior art, with respect to the full cycle of system's architecture, balancing effort of analysis where:

1) Performance Monitoring is used, on an ongoing basis, as a tactical process to detect and isolate performance problems for quick fixes or as an alert for conducting performance modeling (optimization);

2) Performance Modeling is used, on an on-demand basis, as a strategic process targeting optimization of system's architecture.

This hence promotes equal usage of these critical processes as needed, offering a complete solution for enterprise Clients with the aim of providing a high accuracy performance engineering analysis and capacity planning.

In addition, there is a need for a method and apparatus for converging practices of software-engineering design with 'production' system/network administration, when feasible during system life-cycle, as per computing needs reflected by the dynamics of the business process the system supports.

Another long felt need in the industry is to allow the essential understanding of how systems' architecture is utilized with relation to the business processes they support; to have 'just in time and just right' structured approach for maximizing investment into the system and managing its resources .

STATEMENT OF THE INVENTION

An aspect of the invention provides a method for performance monitoring and analysis of a system, comprising constructing a synthetic workload model of the system, comprising selecting a parameter indicative of system performance; monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run; constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM; constructing a system state model (SSTM) associated with the SSM, the SSTM representative of the state and static positions of the SSM; extracting a system performance model (SPM) from the SSTM; and applying a predetermined set of system conditions to the SPM to model and thereafter monitoring a point of failure of the system performance.

Another aspect of the invention provides an apparatus for performance monitoring and analysis of a system, comprising means for constructing a synthetic workload model of the system, comprising means for selecting a parameter indicative of system performance; means for monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run; means for constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM; means for constructing a system state model (SSTM) associated with the SSM, the SSTM representative of the state and static positions of the SSM; means for extracting a system performance model (SPM) from the SSTM; and means for applying a predetermined set of system conditions to the SPM to model and thereafter monitoring a point of failure of the system performance.

Another aspect of the invention provides an method for providing an interface for a user to construct a system performance model (SPM) from a system state model (SSTM) representative of a system for performance monitoring and analysis in a modelling language (ML) environment, comprising selecting a modelling method with a method selector for extracting the SPM; and extracting the SPM from the SSTM with a modelling editor, the SSTM representative of the state positions of the system for the modelling editor to use for extracting the SPM, extracting the SPM from the SSTM based on a selected modelling method.

Another aspect of the invention provides an apparatus for providing an interface for a user to construct a system performance model (SPM) from a system state model (SSTM) representative of a system for performance monitoring and analysis in a modelling language (ML) environment, comprising a method selector for selecting a modelling

method for extracting the SPM; and a modelling editor for extracting the SPM from the SSTM, the SSTM representative of the state positions of the system for the modelling editor to use for extracting the SPM, extracting the SPM from the SSTM based on a selected modelling method.

Another aspect of the invention provides a method for performance monitoring and analysis of a client system in a network computing environment, comprising providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, receiving at computing center a service request from the client system for performance analysis of the client system, the service request posted from the computing center to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, , a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

Another aspect of the invention provides a method for performance monitoring and analysis of a client system in a network computing environment, comprising providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, receiving at network interface a service request from the client system for performance analysis of the client system, the service request posted from the network interface to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

Another aspect of the invention provides an apparatus for performance monitoring and analysis of a client system in a network computing environment, comprising: means for providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, means for receiving at computing center a service request from the client system for performance analysis of the client system, the service request posted from the computing center to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

Another aspect of the invention provides an apparatus for performance monitoring and analysis of a client system in a network computing environment, comprising means for providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, means for receiving at a network interface a service request from the client system for performance analysis of the client system, the service request posted from the network interface to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

Another aspect of the invention provides a method for performance monitoring and analysis of a system in a network environment, a client connected with an application service provider via a network the method comprising receiving from a client a service request for performance analysis of a system; conducting system performance analysis of

the system upon receipt of the service request, the analysis performed with a unified systems modelling process and system monitoring process; preparing a report of the system performance analysis of the system to the client for a fee.

Another aspect of the invention provides a method for performance monitoring and analysis of a system, comprising constructing a synthetic workload model of the system, comprising selecting a parameter indicative of system performance; monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run; constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM; and applying a predetermined set of system conditions to a system performance model (SPM) to model and thereafter monitoring a point of failure of the system performance.

An embodiment may provide a method and apparatus that provides refined granularity of the level of analysis using client system's (the system analysed) architecture raw data in technical format such as system/application resources metrics, system/transaction logs, etc., and transforms by algorithms to provide a result that is manageable and useful data modelling the representation of the analysed system. Enabling modelling with monitoring converging into one framework, combining several modelling disciplines of simulation/analytical/numerical into a single interface, application service providers collaboration, asynchronously and autonomously, engaging in solving client's technical issues that require fine-granularity system performance technical analysis.

Other embodiments provide a method and apparatus that identifies worst case scenarios (WCSs). The method comprising by: (1) initially analyzing subject-matter system data (i.e. system resources metrics, transaction/system logs, etc) as per global system key performance parameters set, verify for completeness; (2) then, using pre-defined system key system performance set then analyze system's worst case time slots in each parameter from this set, by using a define search time period of constraint parameters.

Other embodiments provide a method and apparatus that model reverse engineers using multiple search process as per synthetic workload model for each identified WCS from WCSs set. A method comprising by using trace data construct SDM; if data is available, then construct per each WCS from WCSs set an SSM object; mapping using interim data tables, mapping is per objects mapped in SDM, using data set in interim data tables, to construct each SSM and its system state model (SSTM), while keep links to SDM by objects links. The step of mapping may be repeated until all SSMs have been built.

Other embodiments provide a method and apparatus that constructs PoFs for each interim PoFi (i = 1,2,...) is resulted of calculating an average value as per current evaluated system performance parameter (e.g. CPU Queue) in relation to a defined system resource constraints and overall response time as per currently evaluated SSM from the SSMs set; (2) once in simulation run, of the SPM, if the simulated system reach response time above define value (as define in SDM (default value set to current average plus 1-3 sigma by PMF's Utilization functions)) or reach limitation in system resource as defined then at this point it mark the level in system performance parameter as a PoFt. Another embodiment may provide a method and apparatus that constructs Final PoFs by using model verification techniques then calculating the probability using state data in SSTM and SPM with data of system overall run time then for each PoFt calculating such probability; resulting with PoFs set. Another embodiment may provide a method and apparatus that enables

prediction as per any design type variation of subject-matter system-architecture's and its associated point of failure set (PoFs) as per its current state and the system computing limitations; it provides a critical information allowing 'just in time' and 'just right' system's architecture design management; A method comprising: (1) Via UPMF's Experiment Analysis functions creating a new what-if model by create new copy of live PoFs' associated with models with trace feed connection; (2) made design update in SDMs (and in their SSMs); (3) calibrate and re-run SPMs and further steps to mark PoFs as baseline1, baseline 2, etc; (4) measure PoFs as per baseline point across of PoFs until reach optimum PoFs set (OPoFs), i.e. OPoFs set is defined as system's architecture with the lowest cost and with high 'mileage' design configuration as per business processes requirements it supports; this method is through UPMF's Experiment Analysis functions.

Other embodiments may provide a method and apparatus that enables optimization of subject-matter system' usage patterns (by users or batch workload) by iterative optimization using their associated PoFs data, identifying OPoFs set by: (1) trial and error of system's workload patterns adjustments, by shifting system's usage by users/batches and review PoFs results, or (2) using copy data as received by conducting analysis SPM, edit trace data prior to re-calculation of SPMs, then re-run PoFs calculation and compare PoFs per each baseline until identify OPoFs set, this method is avail via UPMF's Experiment Analysis functions.

Other embodiments may provide a method or apparatus that visual and interactive aid PoFs Insight technical feature enabling to visually and dynamically review System's architecture's PoFs data in conjunction with their model (SSMs, SMDs) and other performance metrics, e.g. response times, SLA, QoS etc. This technical feature is using to generate PoF as well as what if analysis, also used at any point during the system or component life cycle locally or remotely in a network environment; its hence simplify

performance review activity by reduction of performance data, while increase quality and completeness of performance scope; also, it enhances the scope and cost effectiveness for Visual aid PoFs Insight of subject-matter systems supported, by allowing this feature to perform from one interface as per systems' architecture which has dynamic or fixed configuration.

An embodiment provides a method and apparatus that enables grid resources automatically load balancing by enabling: to automatically load balance computer centers with the criteriaof computing centers' system-resources availability, on-going or on-demand service jobs contract constraints, logs of modeling/monitoring jobs run-time estimations and derived priorities and data relating to current jobs running in the computing centers; it solves the technical problem of balancing Grid' Computing Centers resources maximum usage while meeting service contract constraints; and uniform routing of exceptions. Another embodiment provides a method and apparatus for computing & communication Grid structure feature; that enables technology analysis process asynchronously; it solves the technical problem of balancing Analysts and Computing resources in order to provide an economy of scale solution for Clients in technical analysis based on on-going or on on-demand environment; this is by using dual systems: Collaboration System and Computers Centers Systems. Another embodiment provides a method and apparatus that performs asynchronous and uniform exception handling based on a Grid platform technical feature; in performance engineering service environment it enables 1) SLA/QoS/PoF's, and 2) used in log-in services for routing services request between grid parties, exception to be routed automatically to SPs based on exception type criteria, origin and service-contract terms. Another embodiment provides a method and apparatus that enables concurrent computing of large number dynamic system performance modeling with monitoring of subject-matter systems' architecture based dynamic configuration (e.g. Agent based systems) and fixed

configuration (e.g. client server); it enables cost reduction of such technical process while maximizing versatility when using the same technology for any type of configuration.

BRIEF DESCRIPTION OF THE DRAWINGS

A method and system incorporating the present invention will now be described, by way of example only, with reference to the accompanying drawings, in which:

FIG. 1 shows a grid network transposed on a world map of collaboration system in accordance with an embodiment of the invention;

FIG. 2 shows a schematic block diagram of a system in accordance with an embodiment of the invention;

FIG. 3 shows a table listing a system of solution system platforms in a single integrated solution across three layers in accordance with an embodiment of the invention;

FIG. 4 shows a time-line of steps of a method in accordance with an embodiment of the invention;

FIG. 5 shows a schematic block diagram of a system in accordance with an embodiment of the invention;

FIG. 6 shows block diagram of a model blocks and corresponding block functionality of a system in accordance with an embodiment of the invention;

FIG. 7 shows block diagram of a model blocks and corresponding block functionality of a system in accordance with an embodiment of the invention;

FIG. 8 shows a grid layout of a grid system in accordance with an embodiment of the invention;

FIG. 9 shows a schematic block diagram of an apparatus in accordance with an embodiment of the invention for performance monitoring and analysis of a system;

FIG. 10 shows a graph of worst case scenarios for selected parameters in accordance with an embodiment of the invention;

FIG. 11A-B show graphs of deriving parameters for worst case scenario selection in accordance with an embodiment of the invention;

FIG. 12 shows a grid layout in accordance with an embodiment of the invention;

FIG. 13A-B show protocols for component communication with the grid layout of FIG.12 in accordance with an embodiment of the invention;

FIG. 14 shows a table of subscription and charging of services in accordance with an embodiment of the invention;

FIG. 15 shows a graph of subscription and risk-hedged contract of a method in accordance with an embodiment of the invention; and

FIG. 16 shows a graph of periodical risk-hedged contract price corrections of a method in accordance with an embodiment of the invention.

DETAILED DESCRIPTION

A technical novel solution by system and method, which solves long felt need by the industry for focused, flexible, efficient and cost effective performance engineering analysis solutions. All technical and process components included are enabling a large scale performance engineering analysis (PEA) as complete framework; and increasing value of

refined performance engineering analysis for the end user, then of current art, by its novel technical features and enabling interface to exist performance engineering tools and tapping on performance analyst expertise available.

An embodiment of the invention for processing data represents the physical entities of a computer's hardware and software components. It affects the efficiency and cost-effectiveness of a computer in relation to the business process its supports, by managing the technical process of the computer performance engineering analysis process. It contains records on the computer resources metrics and transactions and transforming them into meaningful data be to be re-configured and re-parameterized computer for better efficiency; as well, it provides design insight enabling better technical design decisions as per architecture fit to the business process it supports.

Innovation is implemented by its computing and communication Grid enabling a performance-engineering analysis services framework. All embodiment's technologies complement each other as sole comprehensive solution; nevertheless such technologies or process can be implemented independently as stand alone. Its Grid implementation was designed as such in order to overcome various technical problems: (1) maximizing a corporations' data security by loose-coupling their subject-matter systems to the embodiment Apparatus; (2) equally, eliminating PEA's tools counter-productivity, if processed at corporate Clients site; (3) enabling collaboration between PEA's service providers; this increases PEA value for finer granularity of analysis together providing a complete analysis as per Clients' system architecture performance capabilities.
The innovation introduces a novel method and system implementation offering a balancing effort of subject-matter system' architecture analysis, during system's life cycle, where: (1) performance modeling with monitoring (Point of Failures technical effect) is used, on on-

going basis, to detect and isolate design configuration/parameterization problems for quick fixes or as an alert for conducting system's architecture design iterative optimization by performance modeling; (2) modeling is used, on on-demand basis, as an iterative optimization process targeting optimization of system architecture's design. Conducting such analysis is beyond human capability.

System and Model Embodying the Invention:

An embodiment provides novel solution to business analysis services demand, without the above listed disadvantages, would preferably allow housing all three performance analysis components under one control environment, namely (i) monitoring/measurement, (ii) simulation, numerical and analytical modeling and (iii) technical analysis collaboration. It also enables separate service entities to participate in providing their own added value services that together provide a complete analysis services product. If such a solution is provided remotely then performance counter-productivity issues are eliminated.

Apparatus scope of an embodiment is: monitoring and analyzing the performance of enterprises' 'system' (e.g. transactional or web-based (distributed) application, IT infrastructure, etc). Specifically, this includes IT performance monitoring, performance modeling and the performance analysis process. System's objective is to provide cost-effective performance analysis services to any company that would be interested worldwide. These services can be subscription and risk hedged contract based and offered at pre or post clients' 'system' deployment stages. 'System' resources would hence be optimized offering effective support to clients' business operations while maintaining a scaleable and performance risk-free 'system' operations support for the business process cycle.

One method of an embodiment comprises receiving 'system' measurement metrics and 'system' architecture model data at system's designated web site. Data will come from multiple hardware-devices/software-components that together will comprise the 'system'. Automatic monitoring is by identification of abnormality failure events or patterns in 'normal workload' patterns. Statistical analysis methods can be used. The system provides summary reports with highlights on areas of interest where further action/performance-analysis is required. It provides automatic alerts for areas where performance shortfalls are expected. When performance shortfall has been identified or a subscriber has initiated performance service request then performance analysis-modeling is conducted. The aim is to identify a cost-effective and optimal solution from the set of solutions available in effort to pro-actively rectify the performance issue.

Apparatus implementation of an embodiment is based on:

1) Web-Service's technology that allows subscribing enterprises as well as 3rd party service providers to communicate via cluster of the service provider's Web-Services enables sub-systems worldwide.

2) Performance monitoring and modeling algorithms to be performed on Computing-Centers worldwide and geographically spread. Computing Centers are designed as 4-tier Web sub-system architecture.

3) Productivity related sub-systems that support the efficient and automated operation of the system.

A method of an embodiment is:

1) Service Request workflow based on on-going or on-demand clients request.

2) Entities participation based: Enterprise's Clients, 3rd party service provider, Service Provider's Local Offices Analysts, Service Provider's Back Office Analysts, Computing Centers and Analysis Content Brokers.

An aim of an embodiment is to provide performance monitoring and modelling services to most enterprises worldwide at an affordable price. Another embodiment is to provide product concept available providing an enterprise subscription based performance analysis services by combination of monitoring, modelling and reporting does not exist in the current state of the art.

System's Point of Failures (PoFs) Technology

That system modeling tools, such as BMC's Analytical Modeling or the application Simulation from Hyperformix Inc. of Austin, Texas, United States of America, etc., are rarely used because it is difficult to model a complete system in a unified environment, i.e. different tools and modeling techniques are needed for different hardware / software components configuration or different stages of system life cycle (with or without trace data). Such tools are expensive and require skilled resources to operate. One cannot extend modeling methods within one interface rather then use another modeling tool. Hence current modeling solution cannot be scaled-out. One cannot follow one methodology as per modeling when defined and run a model, hence incurring higher labor costs. It isn't feasible to conduct modeling with close relation to performance monitoring.

To increase modeling automation and methodology in order to achieved reduction of skill labor cost. To enable PoFs calculation by interfacing with Performance Monitoring functions. To enable performance modeling of subject-matter system's architecture based dynamic or fixed architecture, from one interface, using similar modeling methodology. To enable conducting modeling as needed, in line with ad-hoc needs pattern, with low or

no effort needed to construct the model or re-calibrate it. Make such modeling activity a better cost effect solution.

By loose-coupling modeling to modeling implementation (MI) and by modeling in stages, where such stages are managed by one component, allows modeling scale-up from one interface of systems based dynamic or fixed configurations. Also, use of a wizard-tool component is assisting in conducting a consistent modeling methodology by: (1) identifying a problem domain and formulating the problem definition; (2) following by, model formulation with Modeling Editor component; (3) following by, use of a method selector applying the model to a specific method; (4) following by, completing the model definition per selected method (e.g. Stochastic Process Algebra (SPA)). At this stage the model is fine tuned by calibration. Calibration is conducted by reviewing monitoring data populated by interfaces or by creating another model object using the same process by using a different modeling method (e.g. Layer Queuing Network Modeling) and measuring models simulation results like to like. All models once created can be copied for view only or for further modeling for a what-if analysis; it hence allows model object to be created and populated with trace/stochastic data and models can be branched from each other. All modeling is formulated and processed at a designated remote site.

PoFs are defined as system's resources level where the system cannot perform most, for example 85%, of its computing requirements as per the business processes it supports. Currently there is no clarity, existing solutions provide no high accuracy and are very expensive, when trying to conduct system architecture's PoFs prediction, dynamically and with high precision. This is as per subject-matter system's architecture, which are dynamic or fixed configurations based (e.g. Agent based computer system based Web-Service

technology or order system based client sever architecture respectively). The problem origin is that system performance modeling and monitoring processes and their data currently is loose coupling. One can do modeling or monitoring, but cannot do both within one component's interface, as steps from one cyclic and unified process. That current system "probes" can provide data on thousands of parameters within enterprise software and hardware systems, yet there is no clear way to determine how these parameters and thresholds should be used to give a reliable measure of utilization and upgrade requirements.

This lack of confidence originates from lack of essential understanding of how enterprise systems (hardware and software) are dynamically utilized, exactly, in relation to the business processes they support.

In an embodiment, a new paradigm of performance engineering by PoFs, where PoFs are identified and monitored dynamically during system's life-cycle. Enabling adaptive identification and monitoring of PoFs with respect to subject-matter system's architecture design (based fixed and/or dynamic configurations), allowing to maintain up to date system's model and its PoFs. Enabling to measure system's performance, with high accuracy, in term of its systems resources with relation to the systems' architecture design. It hence allowed simplification, efficiency and cost effectiveness in monitoring and optimized system architecture as per the business processes it supports.

In an embodiment a concept is introduced where system's modeling process interlinks with system's monitoring process. It offers intelligent performance monitoring with higher accuracy predictions and better cost-effectiveness then current art. It is based on the insight that both "processes" can be one cyclical process, hence "unified modeling and monitoring

analysis". Modeling data is determined after some data transformations the system Point of Failures (PoFs) set as per systems' performance parameters set. These PoFs set represents a compact set that predicts a system point of failure as per the synthetic workload of the system. The PoFs hence represent, with least data feasible, the system health in its ability to maintain system's workload in correlation to time dimension. It has high accuracy with up to 90-95% statistical confidence, of simulation modeling; as well simulation is synthetic workload based, which results in that a simulation model can be created and processed faster then system complete simulation. The PoFs set enables increased value and efficiency for technical decisions per system's architecture design and maintenance. This is by a simple and pin-pointed correlation of design update and re-configured/re-parameterized to its predicted end-result, enabling 'just in time and just right' capacity planning. It hence solves the technical problem of system performance monitoring current solutions. These typically reports problems too early, resulting in over capacity (i.e. unnecessary expenditure), or too late, resulting in under capacity (i.e. system failure); It is due to their non-linear attribute of their functions and the inherent complexity of their system' architecture model.

The input data used is subject-matter system's resource metrics based, system/transaction logs, etc. This data is analyzed in order to determine key performance parameters and worst case performance as per each parameter. Then by tagging data automatically through mapping components in dynamic construction of system architecture deployment model and worst case scenarios model as per key performance parameters. The representation of these models is standard model language based used in prior art, hence enabling system architects/designers and performance analysts view in a common language. Further models are constructed from the new formed models and system's data; enabling to extract performance model from these models; by using embodiment's Unified Performance

Modeling Framework (UPMF) features. It allows to create a performance model Stochastic Petri Net based, Stochastic Process Algebra, Discrete Event Simulation, etc; this is as per subject-matter system's architecture synthetic workload state positions. Such performance model is linked with its source models and data; it is hence dynamically updated with the synthetic workload updates.

All technical processes are automatic and evolving, resulting in an adaptive view of PoFs and associated models of current status of system architecture. Its adaptability allows viewing and optimizing as per system's architecture configuration changes or business' process workload changes (usage patterns) by optimizing PoFs. It also allows an enhanced drill down system diagnostic by using PoFs data with its associated models, then current prior art.

This embodiment has been formulated so its intelligent performance prediction by PoFs and enhanced performance modeling can be used from one interface to provides performance engineering technical analysis of distributed (dynamic) and non-distributed (fixed) based systems' architecture. As long as system probes are feed into the Apparatus, non-distributed and distributed system's architecture will be analyzed. In cases of very complex distributed system's architecture this embodiment's features can be used for performance engineering analysis of system's agent component rather then the whole system, for example in the case of a bank credit check, the system is connected via Web Services technologies.

PoFs Synthetic Workload Model Construction based WCS

FIG. 10, 11A and 11B show graphs 202, 223, 224 of worst case scenario generation for selected parameters. A method disclosed in an article by Sreenivasan and Kleinman

(S&K), "On the construction of representative synthetic workload." Communications of ACM, 17(3): 127-133, March 1974, offers the view that system's demand is occurred in Xs dimensions of System's resources (S is discrete number of system's resources – hardware and software, e.g. I/O rate); where some resources are more influencing the workload model then other; such influences are marked by weights Wj ($1 \leq j \leq S$) associated with Xi , WjXi. Such demand is represented by Ji ($1 \leq i \leq N$, where Ji is computing Job and N can be very large discrete number) as shown in graph 223 of FIG. 11A. The S&K method is too complex to use: (1) when number of discrete computing Jobs (Ji) is very high; as S&K method was design for mainframe computing where in modern Client Server and Distributing computing there are many Jobs occur concurrently that needs to be included in the workload model; (2) its synthetic workload model (SWM) formulas calculations are too complex; (3) synthetic workload modeling not allows for software engineering practice to use in conjunction with performance monitoring and analysis practice, by allow to have common synthetic model view with properties concern both practices. This embodiment method uses three principals from S&K method of: (1) Xs workload model space of system's resources, where system's demand is occurred; (2) SWM is equivalent (or a good approximation) to real system's workload, if the density probability of synthetic and real workload is the same; (3) SWM is derived from real workload using Constraints.

The embodiment' feature offers improvement on S&K method by: (1) the Constraints are defined in correlation to bottleneck of the system's resource (e.g. system's CPU), where some Xi ($1 \leq i \leq S$) have certain value associated (e.g. X1 ≥ 2 and X2 $\geq 90\%$, where X1 is CPU-QUEUE and X2 is CPU UTIL - Utilization); hence it different from S&K method where Constraint defines as cap on system's resource for the modelling's time period; (2) in relation to (1) above, then values of Xi in conjunction of Xj, Xk ($1 \leq i,j,k \leq S$) can be different then in its conjunction with Xl ($1 \leq l \leq S$). It emerges from above that in Xs

dimensions space of system's resources is closed by the edged form by these Constraints, re FIG. 11B showing the graph 224. All system's Jobs Jj ($1 \leq j \leq P \leq N$, where Jj is computing Job and P can be very large discrete number) outside the edges of the now Xs dimensions closed space, (e.g. marked by bold lines orthogonal to Xs axes in FIG. 11B, are where the computing is with delay factor of the bottlenecks. Equally, all Jobs Jk ($1 \leq k \leq N$, where Jk is computing Job and N can be very large discrete number) within the edges 225 of the closed space are where Jobs performed without bottleneck effect.

Some of the Jobs on the edges of this closed space are part of Worst Case Scenarios (WCSs). Worst case Scenario is when many of its computing Jobs are closed to the edges of this closed space. Worst Case Scenario is continues sequence of Jobs in discrete time; hence we ignore all Jobs that are not related to WCSs, as their effect to cause system resource's bottleneck is significantly low in comparison to those Jobs in WCSs. FIG. 10 shows a graph 202 of WCS 204,206.

The propose method is performed upon Xi ($1 \leq i \leq Sk \leq S$, in comparison to S&K synthetic workload model), where Sk is the number of key system performance parameters; further reduction of synthetic workload model data is expected on by using bottleneck Constraints; lets marked Xsk closed space as X§. Then a search on each such Xi is conducted to identify time period where Xi is above certain define Constraints threshold, e.g., within the search period range. X1 ≥ 2, X1 is CPU QUEUE; this is for a define other Constraint - Time, e.g. T = 30 minutes; this search is for combination of Xj, Xl, etc with ($1 \leq j,l,.. \leq §-1$) as per each edge of this closed space (e.g. X1 is CPU_QUEUE ≥ 2 and X2 is CPU-UTIL $\geq$ 90%, X1 is CPU QUEUE ≥ 1.75 and X3 is Active Processors > = 150). The result of search is a WCSl set ($1 \leq l \leq M$) then we search to match the probability of real (i.e. open space Xs) to synthetic (closed space X§) by selecting a sequence of WCSs with its

probability (e.g. to use for DES, Network Queuing modeling) or use stochastic probability (e.g. SPA, SPN)

Referring to FIG. 9, an embodiment is shown of an apparatus 200 having system scenario model (SSM) 210, system deployment model (SDM) 212, system state model (SSTM) 214, system performance model (SPM) 216. The interaction, components names, object and implementation of embodiments is discussed. A Modeling Workshop Manager component (MWMC) is initiated; then a call is made to initiated a Modeling Editor component (MEC); in MEC a new Model object (of client's) is created or allows to open existing Model object; (a) in case of new Model object, then it open dialog box of selection of: (I) work with trace data or (II) without; then Analyst selected Client System ID (CSID) from Client option of file menu; (b) in case of exists Model object then an Analyst selected Client System ID (CSID) from Client option of file menu; in case of (I) then a PoFs Projection Manager component (PoFPMC) is initiated with passing reference to the Model object; then a call for Worst Case Scenario (WCS) Identification Component (WCSIC) is initiated with passing reference to the Model object; in WCSIC Analyst made selection of System Type list window (e.g. client server base Unix, Web Server based Linux, etc), once a selection is made then UI window is open with list of key system performance parameters (KSPPs) or key component performance parameters (KCPPs) with their pre-define associated weights WiXj , $1 \leq i,j \leq s$, as shown in the graph 202 of FIG. 10; then Analyst is updated this list of WiXj as see fit; then Analyst is open a search constraints criteria UI window, by make selection in main UI menu; the selection is set at default value, in systems parameters, for: (1) one month data from that current date; (2) search period of Monday – Friday (3) search interval of one hour between 8am-6pm; (4) value of Xj with reference to other Xi , e.g. Xj CPU-QUEUE ≥ 1.5 and Xi = CPU UTIL $\geq 90\%$, (constrains criteria can be expanded in options in the system's parameters); then Analyst is made amendment to the Model's search constraints criteria as see fit; this search is save as

store procedure in data-warehouse; then a job request is made to Monitoring-Exception Scheduler component (MESC) which interact with Grid's Optimization Resource Manager component (GORMC); on selection of job type (i.e. WCS search) Analyst made selection of: schedule time to perform calculation, priority, reference to store procedure object; once store procedure initiated by GORMC then search is conducted on all data as per the selected CSID in the CC's data warehouse; selection is for highest mean value of Xj with constraints set before, while following synthetic workload existing principles; each such selected time period is set to intermediary table of time period as identified in the search, e.g. two WCSi selected per X1 measurement data of Unix server - CPU-QUEUE, re FIG. 10, now a search is conducted on all said system/transaction logs, etc and build in the intermediary tables the data allocated for that time period; this done for all WCSj identified as per all Xi, WCS objects are contain reference to its associated intermediary tables; on success then WCSIC return reference to Model object, otherwise return error message to PoFPMC; then PoFPMC return to Modeling Workshop Manger component (MWMC) with reference to Model object; MWMC is then initiated Worst Cases Scenarios Models Construction component (WCSMC) with reference to Model Object; WCSMC is then load intermediary table set before and for each WCS object of WCSs, at initiation it construct the objects: SSM, SDM of unified modeling language (UML)-Model object; SDM 212 on constructing is call a method with reference to Model object to mapped architecture into system deployment model based UML, this by extract from XML format made by 3rd party tool (e.g. HP OpenView Net-Manager); such data is updated at data warehouse on frequency set in System's parameters (e.g. every day); the extract to SDM view is using 3rd party tool; then SDM object UI method is open the SDM UI window by call to method from reference of Enterprise/Application Architecture Mapper component (EAAMC) via MEC; Analyst made update to SDM view model as see fit; then search is made to all items in intermediary table as per WCS time period; then tagged all items in the table and then

using recursive function search of link between tagged items to each other; per each pattern links then check for reference to occurrence as per SDM linked reference; on valid links then checked for tagged item pattern accuracy as per value in System parameter at SSM 210; on error then generate exceptions; if exception raised then analyst call to view all linked logs via call in file menu, logs appeared in UI table window, then Analyst can made manual update of data(debugging) and re-start the process; once process complete successfully then SSMs and SDMs pairs as per WCS are linked, then a SSTM 214 object is constructed; then two objects constructed per SSTM: SSTM Statechart and SSTM collaboration; on construction, object made call to method to auto-generated from SSM object's model the statechart and collaboration model respectively; on exception then Analyst can manually updated SSM's model script via call to MEC's method, otherwise return success; then MWMC initiated Modeling-Problem Formulation Wizard component (MPFWC); the MPFWC object on construction load selection made before or open default selection; Analyst made selection following performance modeling guideline pre-define in script in System's parameters; the selection from UI windows call in sequence once selection is made; Analyst made selection of modeling parameters and guided through structure methodology to select method on some selection an calculation by method call will be made on Model's data; such selection data is save in Model's object; on selecting of modeling method a call is made to Methods Selector component (MSC) which construct via UPMF 220 object as per selection made in MPFWC, e.g. SPA was selected as method which create an object of Numerical-Modeling Methods Management component (NMMMC) where PEPA API method is generate SPM 216 object using SSTM object; on error raised exception or return Model object to MWMC after made call to Performance Analysis Measurement Component (PAMC) which generate job request to MESC to calculated performance metrics as per each Model object's SDM; in case of (II) MEC, via MWMC, using selected CSID made call to construct SDM (re FIG. 9 and 10 1-(3), which

SDM object constructed as per reference to XML format generated by 3rd party tool (e.g. IBM's Rational Rose) and SSM is constructed in similar way to SDM; this process is repeated for all SSMs and SDMs pairs mapped in system data from Client site; then SSTM and SPM per each SSM is constructed in similar way as above; SPM correspond to the state of the selected stochastic process; to generate PoFt then Analyst made call via MWMC to MEC, where Analyst can load Model object using CSID implementing a model verification transformation; a UI window of SSMs and SDM window appears where SDM UI window show data as per linked current active SSM; Analyst can at this point update the constraints at Systems parameters and re-start a new Model (e.g. reduce time period from 1 to 0.5 hours, CPU-QUEUE level from 1.5 to 2.5) or analyst can delete SSMs (and their associated SDMs, SSTMs and SPMs) as see fit; once Analyst have compact set to progress in calculating PoFt then Analyst can start calibrating SPM; this by: via open SDM UI Window (MEC method) then set per each SSMs the SLA/QoS values by call to Monitoring Against SLAs/QoSs/FoFs Targets component (MASQFTC), by making selection in right mouse option; on object constructing it check with 'traced' Model object if performance metrics are existed, if not return an exception; SLA/QoS value are set by interactively set value on SDM per each active SSM from Model object's SSMs set; Analyst can modify model performance parameters by selection an option in SDM's file menu; then Analyst select to run the Model's SPM by making selection in right mouse option; this will generate a job request to Models-Execution Scheduler component (MOESC), which will interact with Grid's Optimization Resource Manager component (GORMC) and schedule time for call store procedure call with SPM value as per SLA/QoS to simulate each SPM; on completion this procedure will initiated a call to PoFs Identification component (POSIC); then in POSIC via 3rd party software API (e.g. Probabilistic Symbolic Model Checker (PRISM) by Department of Computer Science, University of Birmingham) will calculate PoF from each PoFt by using computing

requirement data load with system log; if such computing data isn't available then PoFt is set to PoF in the Model object, as shown by arrows in FIG. 9 PoF calculation 218.

In case of (b) Model Experiment Manger component (MEMC) 104,134 is initiated, which subsequently load the Model Object in its construction, use CSID as reference; MEMC allows visual interactive that allows model copy (by construction a new model object), and update this object data (SSM, SDM and their associated SSTM and SPM); it allows using different modeling methods, as described above; MEMC provides the What-If modeling analysis by making modification to raw data (usage pattern data, effect SSMs) as per WCS, system configuration (effects SDM) and SSM and review PoFs updated predictions, as shown in FIG. 6 and 7. FIG. 6 and 7 show a block diagram of model blocks 70,100 and corresponding block functionality 170,130, respectively, of an embodiment of the invention. In FIG. 6, models and corresponding functionalities shown are: Monitoring – Execution Scheduler 72,172; Data-Warehouse Services Platform 74,174; Forecasting 76,176; Resource Share Management 78,178; Performance Analysis Measurement 80,178; Enterprise/Applications Architecture Mapper 82,182; Monitoring Against SLAs/QoSs/PoFs Targets 84,184; Operation Exception Management 86,186; Web-Services Platform 88,188; and System Platform 90,190. In FIG. 7, models and corresponding functionalities shown are: Service Request Manager 102,132; Model Experiment Manager 104,134; Modeling Editor 106,136; Models Execution Scheduler 108,138; Modeling Workshop 110,140; Modeling Problem Formulation Wizard 112,132; Methods Selector 114,144; Simulation Modeling Management 116,146; Analytical Modeling 118,148; and Numerical Modeling 120,150. It will be appreciated that other embodiments may comprise of different models in other arrangements.

Operation Exception Management component (OEMC) 86,186 is initiated by store procedures at the data-warehouse, when SLAs/QoSs/PoFs value reached there define thresholds; SLAs/QoSs are set in MASQFTC, re above; PoFs are monitored in Resource Share Management component (RSMC) 78,178, which match between values of Model Object resource object PoFs and its short term forecast values (set by prior initiation of Forecasting component (FC) 76,176, which is run minimum every 24 hours or otherwise as define in Apparatus parameters); RSMC is batch based job, initiation as defined in Apparatus parameters; OEMC set values for exception record set by Grid Technical Analysis Protocol – Management (GTAPMC) component, re FIG. 6;

The implementation in accordance to an embodiment is specific UML environment as per 3rd party tool support use through this embodiment's PoFs process for Object Management Group, Inc. (OMG) unified modeling language (UML) 2.0 standard. However the concept of this embodiment is supportive of any modeling language with capability for: (1) modeling activities/scenarios (e.g. UML sequence diagrams) flow; (2) static (e.g. UML Collaboration Diagrams) and stat flow (e.g. UML Statechart Diagrams) that needs for performance modeling; (3) industry standard that use for performance analyst and software development professional to co-interact as per common view of the system's architecture.

The system may be dynamic or fixed. For example, the following scenarios:

(1) synthetic workload model uses for modeling and monitoring is of 'SYSTEM' demand on key system's resources, within a close past time period; any SYSTEMi configurations of 'SYSTEM', constructed and performed within this past time period, is performed demand on some of 'SYSTEM' key system's resources; hence the overall health of 'SYSTEM' can be good approximated to overall health of SYSTEMi, by measuring PoFs;

(2) in case of: (I) very large 'SYSTEM', where many different SYSTEMi can take place (e.g. Internet as a 'SYSTEM'), then clearly the PoFs method and Apparatus will not provide good approximation for SYSTEMi's PoFs prediction; however the embodiment can use for prediction PoFs of SYSTEMi's component, i.e. for each sub system participate in dynamic interaction of other sub-systems of SYSTEMi; hence by maintaining for each SYSTEMi's component, which is a system by its own, its SLA/QoS and PoFs then the SLA/QoS/PoFs for SYSTEMi are maintain – 'weakest link'; (II) 'SYSTEM' with high variation of SYSTEMi configuration, then component's PoFs prediction based is provided a good solution for reasons stated above.

Grid's Performance Technical Analysis Technology

Performance Application Service Providers (PASPs) Process Technology

Currently it is very expensive and complex to have subject-matter system's architecture design management, during system's life cycle stages, closed-coupling with system's monitoring and change-requests. This is as per alerts initiated pro-actively or near-real time of performance monitoring or change requests based business dynamics respectively. Furthermore, the same apply when subject-matter system's architecture design management, during system's life cycle stages, is to be reviewed and recommend updates potentially by several entities during asynchronous performance technical analysis.

A new concept of performance engineering process, by method and technology, enabling a solution catering for different types of Service Requests (SRs) , i.e. SRs are on-going based as in performance monitoring or on-demand based as in capacity planning (performance

modeling); as well a solution catering for different types of service packages as per different usage patterns. As the only viable option of performance servicing is remote analysis; the best implementation of remote performance-analysis solution is the one that balances solution's system resources (servers, data storage, etc) with performance-analysis analysts' resources worldwide. Performance engineering by service needing to cater for the magnitude of several thousands of SRs concurrently, where performance modeling/monitoring algorithms need to perform under time constraints as stimulated in the Service Level Agreement (SLA) of SPs with Clients or as under specific contract terms between these parties. SPs' performance analysts need to engage in servicing SRs in an efficient way, with little management overhead of analysts' resource allocation and administration. Maintain maximum security to client data and access to its system.

It was the insight perceived that the expertise required to: configure, model, monitor, and conduct other technical analysis using performance engineering tools, while using performance engineering methodology is economically beyond reach of mid-to-large size corporations. It is the case even with the efficiency and cost effectiveness of embodiment's other features.

Furthermore, it was the insight perceived that monitoring is an on-going process while modeling (e.g. capacity planning) is an ad-hoc process. So if the technology is able to conduct large scale modeling and monitoring then increasing the technology reach so it becomes ubiquitous technology a collaboration technology has to be developed enabling economy of scale technology for on-going PoFs' prediction with monitoring capabilities and ad-hoc modeling by what-if analysis. Hence a grid solution, of commuting and communication, has been considered.

The Grid was constructed from two connect type of systems: (1) Computing Center apparatus as shown in FIG. 2 (servicing Clients based geographical region as shown in FIG. 1), where the modeling and monitoring related algorithms are processed; (2) Collaboration apparatus as shown in FIG. 5, that provides the interfaces to Computing Centers, Clients, 3rd party providers and PASPs, and the like.

Computing Centers design is based on the insight that the modeling task can be significantly reduced if it caters for small scale system or a system's component. For example, a computing Grid can be only modeled using distributed simulation modeling where model complexity is many factors higher then simulation modeling of system's components that together make the Grid. This insight is perceived from the notion that in distributed computing environment a system's architecture 'component' is managed/analyzed locally rather then the same as per system's architecture as whole. The benefits of such design is dual: (1) modeling with monitoring based small scale system (or system's component) with high accuracy and with software design management focus; (2) significant simplification of modeling with monitoring effort; for example complex distributed simulation modeling is avoided, while scaling up modeling effort can be done with ease, efficiency and cost-effectiveness.

Referring to FIG. 2, Computing Centers were designed as 4 tiers web-servers and application-servers farms with scalable technology. Computing centers are allocated based on geographical regions as shown in FIG. 1 and all computing centers are connected to PASPs Process apparatus via a collaboration system as shown in one embodiment in FIG. 5. Computing centers allocation is subject to best balance of solution resources key factors (servers' farms, data storage, good communication link availability and their cost, etc) with PASPs' analysts resources worldwide. FIG. 2 shows schematic block diagram of the

arrangement 12 for computing centers in accordance with an embodiment of the invention. Service Provider Analysts 14,16 and Service Provider Web Services Servers 61 access the Internet 54 via firewalls 53 to H/W Load Balancing Units 52 to Web/Pulling Servers Cluster 58, Application Servers (Algorithms) Cluster 57, Object Persistent Layer 56, Data Warehouse Server 59, and SAN Data Storage 55.

Referring to FIG. 5, Collaboration apparatus is a 4 tier web-server and application server farm that enables connectivity between entities participating in this process: Clients, 3rd party who manages Clients systems and service provider (SPs). Collaboration systems also are designed to: (1) manage automatically all SPs analysts' resource allocation and administration; (2) enable billing and SR workflow; (3) automatically load balance model and monitor jobs requested at computing centers; in the effort to optimize computing centers computing with the constraint of SLA or contract commitment with Clients. FIG. 5 shows schematic block diagram of the arrangement 18 for Collaboration apparatus in accordance with an embodiment of the invention. Clients 51 and Clients/Service-Provider and 3rd Party Service Providers 62 access the Internet 54 via firewalls 53 to H/W Load Balancing Units 52 to Web Servers Cluster 58, Application Servers Cluster 57, Object Persistent Layer 56, Data Persistence Server 60, and SAN Data Storage 55.

Grid Protocol

It is a fundamental technical problem when conducting performance engineering analysis of an application by modeling with monitoring is the technical issue of black vs. white box modeling with monitoring due to the diversity of software and hardware components being used in the market. It was the insight that by enabling several entities within the modeling with monitoring analysis process as per subject system' architecture, one can overcome such technical limitation. Equally, it has emerge from the view, that system's architecture

related technical analysis can take different routes, as per diagnostic analysis, once exception has been triggered by dynamic monitoring or when technical analysis based what-if modeling (e.g. capacity planning) is initiated. It was the insight perceived that in this embodiment's computing and communication Grid any form of technical analysis process can automatically take place if analysis-process protocol would be available to initiate and manage such technical analysis as per any subject matter systems' architecture.

A new paradigm of ad-hoc technical analysis based Grid, where analysis is autonomously and asynchronously, allowing finer granularity of analysis as per subject-matter systems' architectures on on-demand basis. In asynchronous environment where Performance Analysis process is provided on ad-hoc basis by SPs, where analysis report is automatically generated by the SP's systems or their Analysts comments. Such technology benefits are automatic, efficient and cost-effective management of Service Request process by PASPs.

The complexity is raised when SPs are servicing several thousands of client's systems concurrently, hence there may be thousands open SRs. Efficient servicing cannot be solved by load balancing SRs, by the SPs, or use ad-hoc workflow (e.g. IBM Notes tool, by International Business Machines Corporation (IBM) of Armonk, New York, United States of America) used manually by the Main-SP to control SR servicing with other SPs (or manually between SPs) as SRs may be serviced by application rather then an Analyst.

It is the notion that in asynchronous computing and communication Grid environment, ad-hoc workflows can take infinite routes, according to need to resolve technical process reasoning. For example, in a technical process where several problems have been identified, for one subject-matter system's architecture; the analysis can take place (for each problem), automatically by dispatching randomly to the first analysis service provider

conducting analysis, who may pass it randomly forward to next analysis service provider or solve it and record it in analysis report; also, it may require several analysis service providers to work in conjunction in order to produce (as agent-based systems) the technical analysis needed. Hence a technical analysis protocol based Grid was designed with design simplicity and low overhead computing factors, allowing complex interaction between SPs. It is the notion perceived that if such technical analysis process's protocol does not become industry standard, all systems' architecture under review can still be managed and optimized by the subject-matter features of this embodiment.

FIG. 1 shows a grid network transposed on a world map of collaboration system in accordance with an embodiment of the invention. This example shows only local offices, i.e. service providers local office 12, and computing center 14 entities. Other entities such as clients 3rd party service providers, background analyst sites, etc., are not outlined, mainly because of graphical representation consideration. Also collaboration connection lines are representing as centralized from London, England; however, collaboration connection may be enabled from any entities pair within service-entities set. Additionally, FIG. 8 shows a grid layout 10 of a grid arrangement in accordance with an embodiment of the invention. In FIG. 8, computing center 12 and service providers local office analysts 14 are shown together with service providers background analysts 16, collaboration arrangement 18, and third party service providers 19.

Problem ID 240 and exception protocols 260 are shown in FIG. 13A and 13B, respectively. FIG. 12 shows a grid layout 230. These protocols are used for exception communication and handling, as well as for ad-hoc analysis. The Grid is a platform that managed the demand rises by SLAs/QoS/PoFs exceptions as per clients systems and ad-hoc analysis requests. Exception and ad-hoc analysis requests can take a sequential of pattern form of

interaction between SPs; pattern are differentiated from sequence in that the analysis based pattern is of specific tasks order set from the outset (e.g. calculation Log(a)+Log(b) can be done by: SPi -Log (a), SPj – Log(b), SPk – SPi result + SPj result).

Exceptions and ad-hoc analysis requests are always associated with Computing Center (CC). As exceptions are raised at the CC and ad-hoc analysis is concerned client system located at the CC. The CC details are needed for monitoring the exceptions/ad-hoc requests that has not processed within time-limits for reference only; SPs on submission comments on Client system are sending problem massage with confirmation on success, in S/F part in problem ID.

Exceptions and ad-hoc analysis requests 'demand' within the Grid is initiated at any point in time and each exception/request can be dealt by SPs in their own time, i.e. asynchronous processing. Hence it is vital to put time limit in the exception protocol to flag out those cases which did not asynchronously process within the time limit 270. These cases are dealt manually by designated SP, whose details listed in the Problem Escalation ID 272 in the Exception record. However, an embodiment method has been design to process very large number of exception/requests (SRs) automatically to the needs of technical analysis problem solving within a Grid's of clients and SPs.

The protocol devised as such that there is no need to monitor exception or ad-hoc request. If SP's Analyst or System is failed to process a request then error will raised at designated SPf when time limits expired with no resolution and.exception / ad-hoc request will pass to SPf's Analyst via Problem Escalation ID;

Exception protocol cover exception data: (1) Key (CC ID+ Exception ID), which is unique for time expected to resolved the Exception; (2) T/F 266 for Text or Formula, which is indicator for those SPs that their Analyst/System can process exceptions/ad-hoc analysis requests, while using the same SP's ID; (3) Description ID is give reference to text data to be load or if it is a formula it give parameters to be use by SP's System; the actual description is accessed from exception-description table at database using key of: CC ID 262 + Exception ID 264 + Description ID 268; either way (i.e. text or formula) the details of *Client System ID (CSID) (2 byte code)* are listed at the start of Description details; (4) time limit as advise above; (5) Problem Escalation ID is used when time limit is expired it route error to Analyst/Entity for manual resolution; this field is checked every time SP's Analyst/Systems has received Problem record and load the its exception data; if Problem Escalation ID is equal to SP's own ID then a manual escalation process is initiated; it may able to make a call to interface of support software (e.g. Remedy Product by Remedy Corporation of Mountain View, California, United States of America).

Problem protocol routing data: (1) key (CC ID 242 + Problem ID 244), which is unique for the time expected to resolved the associated exception; (2) Skill Set 246 which categorized the type of problem solver skill needed (problem can be technical analysis based or administrative based for some of logging SRs); this field is used for forwarding the problem to anther addressee based of type of problem solving skill, where the addressee selected in random; for example, this field is used when SPi required to forward problem to another SPj, by making random selection of SP based on skill set field; or is used for logging SRs by clients; (3) Problem Originator ID 248 is of the lead SP who overseen the problem solving; this field is used: (I) ID (address) of SPi to return result that SPj has requested to perform; (II) to transfer overseen responsibility between SPi to SPj as per problem solving, by over-writing the SPi (originator) ID with of the SPj ID; (4) return of

success (failure by default) as per problem; this field is used in the following scenarios: (I) CCi initiated new problem to SPk and SPk solve the problem and made comment to client by inserting new record in table using key: CC ID + Exception ID 252 + respond from SPj or 3rd party SPw that eventually solved the problem. It is hence allows by CSID (such record is removed as define in System's parameters) to be added, at later stage, in a report send to client; then exception record is removed using CC ID + Exception ID; S/F 250 set to S – Success; (II) SPk forward it SPj with no intention to resolved it, i.e. Problem Originator ID is changed from SPk to SPj; S/F remain at default position(i.e. F for Failure); (III) SPk solved the problem as in case (I)...; but, create new problem, which can be processed: (i) SPk request SPl to solved a problem and make SPl as in case (II) to overseen the problem; S/F field remain in default position; (ii) SPk request SPl to solve a problem and return result back, which in this case SPk may received return Problem with his ID and S/F field is with the value of : (a) default (i.e. F), which in this case SPk may initiated a call to other SPg; or (b) success (i.e. S); in case of success then SPk load SPl comment from table using key: CC ID + Exception ID + CSID ID; (5) exception ID used for reference to exception record combine with CC ID field.

Pattern takes place when problem solving by SPk is needed to take shape in certain format (e.g. log(log(a)+log(b)). The routing protocol is always maintained that request from SPk to SPj will result in SPk to break a problem to small problems and structure the routing according to a solution pattern of choice.

The exception protocol is implemented in an embodiment with reference to logging SRs, as discussed with reference to FIG. 13B. SRs logging process is initiated by invoking Service Request Manager component (SRMC). The logging is initiated by the Client's/Analyst's Web-Browser and completed when a report is submitted to the

Client/Analyst respectively. All SRs are report related service-requests. Such reports contain model analysis with description of SDMs and their SSMs with detailed description assisting drill down fault diagnostic follow-up or technical decision; or details associated with escalation of issues by SP's Analyst/System (re Exception protocol - Problem Escalation ID). All reports are compiled by SPs' Analysts/Systems and based on: (1) analysis completed as per Clients original/modified models (SSMs and SDMs based), generated via Performance Modeling platform; (2) analysis completed by SPs own systems, where their analysis is with respect to Client's SSMs and SDMs; SRs related records are inserted / communicated by initiating a call to grid's protocol.

Business Technology Analysis (BTA)

Current Business Technology Analysis (BTA) (e.g. Performance Engineering Analysis) solutions and current method of providing BTASs does not allow separate companies to collaborate together in providing such services for enterprises seamlessly and on full outsourcing terms from anywhere. Hence, BTASs cannot be commoditized.

These services can be sold and managed as a commodity as per their on-going or discounted on-demand service, but any other on-demand service will be charged at full rate. The aim is to provide BTA business method by monitoring and modeling services for targeted enterprises worldwide at an affordable price, (e.g. targeting the IT and Telecom markets) with the characteristics of a utility model. Enabling efficiency by clustering service providers by service type on-going or on-demand services; alternatively grouping based interactive and non-interactive services with Clients. Enabling efficiency and simplification of process by reducing administration as per this process, i.e. automatic client administration, Analyst resource allocation and managing administration; this is by

SR's workflow, re PASPs' Process. Offer BTAS – performance engineering analysis services as a subscription service to mid-sized organizations that are not well served by existing software products. By offering the software as a service; hence eliminating the need for its Clients to obtain skilled employees who would otherwise be needed to operate sophisticated analysis, monitoring and capacity planning software, as well as eliminating the cost of purchasing the software itself. This approach will enable mid-sized organizations to benefit from sophisticated analysis, monitoring and capacity planning systems for the first time. This will bring a massive cost and performance benefits to such Clients by reducing the large amount of guesswork involved in current IT/Telecom system design. Offering BTAS – performance engineering analysis services as a subscription service, based packages, will bring much improved accuracy compared to existing software solutions used by large corporations, many large organizations will also be attracted by its service offerings.

It was the insight perceived that Business Technology Analysis such as performance engineering analysis can perform a complete business cycle in the embodiment propose Grid.

It is perceived that all analysis service activity start with an on-going or on-demand service request (SR) from Clients and end with report that is dispatched to Clients electronically an within an agreed period, as define in subscription package (on-going SR) or quote (on-demand SR) with SPs. Service is: subscription, pay-on-demand (POD) or risk-hedge contract (RHC) based. Subscription offer to client services package, which is package that is most close to the client expected pattern of service usage. POD is devised to increase service volume, but its volume expects to be small part of the total. RHC is devised as a mechanism to spread individual clients risk (by clients categories) to overall clients, risk

are of un-predictable (un-budgeted) change requests and system faults diagnostics, above the subscription package usage. RHC terms are subject to cap and other terms to ensure it capability serve its purpose and suitability with its correlated subscription package, i.e. the lower client's subscription package will counterbalance with higher price comprehensive RHC and vice versa. Subscription with RHC method factors are: (1) maximization of efficiency and risk reduction at clients end; with (2) maximization of this Grid's operations profitability end, while maintains customers demand for contracts; these factors are in direct positive correlation with Grid's economy of scale operations.

Analysts' resources are divided into two groups by activity: Interactive analysis with Clients and background analysis. Interactive analysis is performed by service provider(s) staff together with Clients' staff in technical analysis projects locally. A background Analyst is performing on-going analysis by exception (forecasted or raised) and any analysis that can be done remotely. Background analysis can be provided by designated SPs or contracted specialized SPs. Computing is from computing centers, re PASPs Process. In any computing and communication Grid there can be multiple ASPs who provide Background or Interactive Analysis; also it is possible to have two or more types of business technology analysis automatically performed in one Grid (e.g. PASPs Process based IT systems and Car Manufacturing Systems).

For example, such data is directed to the nearest center and once processed its reviewed by center's automatic monitoring process to be tracked immediately or forecast abnormality to its normal workload patterns. If abnormality is identified then it is flagged out to background-analysts (SP) that will conduct analysis or dispatch it to other service providers' teams. At any point Clients can raised service requests via the Collaboration system. Service requests are processed by designated service provider who allocates

Analyst(s) resources from its pool and/or from contractor service providers to conduct analysis on performance analysis project. Any computing needs such as modeling, monitoring or other specialized tools are provided from the computing centre. Analysts teams will interact with the computing centre, as needed, in order to process modeling or other algorithms. An Analysts team may be allocated from a local Analyst(s) to a client and anywhere globally where team members will collaborate via a Collaboration system, re PASPs Process. The Collaboration system provides: SR workflow processing and monitoring, allocated analysts resources per on-demand jobs, automatically optimizing on-going and ad-hoc jobs schedule of computing center's algorithms.

It was the insight perceived that with embodiment's PoFs features, where business technology system failure potentially is significantly reduced, SPs now can be measured up against their performance by their ability to optimize the subject-matter Clients system architecture's PoFs set. In an integrated open business technology service environment, service can be effectively measured up by objective results of optimized PoFs; hence PoFs data can be used as service measurement parameter in BTAs' service 'commodity', for basic services; equally PoFs can be used as parameters for setting service price and service's penalty (if during service, system's PoFs exceed the guaranteed PoFs based service contract) between client or 3dr party companies representing Clients and SPs.

Services are based on on-going and on on-demand. Services are by singular and/or floral SPs. On-going services are: PoFs prediction, Diagnostic fault management and on-demand services are What-If analysis (e.g. capacity planning) and other general performance analysis services by SPs' Analysts. Clients are charged for: (1) Grid's access and administration (billing, etc) which cover by Grid's collaboration apparatus; (2) SPs' Analysts/Systems services, which cover: (i) technical analysis based performance

monitoring and analysis and (ii) support and administration (clients support, Consultancy administration, Client Account Manager); (3) Computing Centers usage as per on on-going or on on-demand SRs; (4) Access via 3rd parties systems (in case clients outsource their systems to the 3rd parties).

Services are charged on basis of usage alone and usage with risk-hedged contracts. FIG. 14 shows a table 285 of subscription and charging of services in accordance with an embodiment of the invention Usage contract are: subscription and pay-as-you-go. Subscription is based 3 packages: base (re Subscription Price -Sliver, e.g. PoFs prediction), improved (re Subscription Price -Gold, e.g. PoFs Prediction and System Faults Diagnostic) and comprehensive (re Subscription Price - Platinum, e.g. PoFs Prediction, System Faults Diagnostic and What-If analysis work-units credit); these packages are calculated on Software (application service providing – CCs plus Collaboration costs and 3rd Parties costs if applicable) and Consultancy (SPs' Analyst/System cost); the price clients are charged is calculated: (1) Software - in term of hardware items to cover the cost of Software (e.g. workgroup tier); (2) Consultancy – multi-skill Analysts cost (and systems cost) to support demand for technical analysis (or support) to cover comprehensive cost of staff/system; subscription charges are set to covered the cost of Grid's operation and add-on service by parties involved in SR's process; hence any revenue on top of subscription revenue is devised as gross profit.
Pay-on-demand (POD) is based on initial setup and analysis cost with the flexibility for Grid's platform and services usage whenever is needed, at rate of cost + profit margin without any discount, i.e. not charge at reduced rate, subscription's packages nominal cost rate; it covers the comprehensive cost of software and analyst for all parties participates in the SRs process; Client will be charge on either POD or subscription contract; POD is

devised to increase service volume, while it is expected that its volume would be small part of the overall volume.

Risk-hedged contract (RHC) are guaranteed fixed cost as per uncertainties of IT, Telecom or Business dynamics; these contracts are offered to subscriber clients only; their value to client is by mitigating risks of un-predictive (and un budgeted!) change requests/ system fault diagnostics as per dynamics external to the on on-going (and some predicted on-demand) of system's operation during its life cycle; such individual client risks with RHC are spread across all clients, with the risk-adverse cost implication for clients; RHC pricing is based on: (1) system's complexity in terms of number of its PoFs, i.e. the work effort complexity estimated; (2) how often the system needed change requests / system fault diagnostics, i.e. the work volume estimated based past data; (3) periodically profitability margin corrections, re below; (4) risk reduction discount – adjustment of average RHC as per reduction of risk in effort to maintain an overall fixed profit margin while encouraging clients to use RHC economically, re below; (5) number of clients; (6) client categories. i.e. type of services; (7) external and administration cost for Grid's service parties; client categories average RHC price is also including consideration for market overall/segment profit margin vs. building client based.

RHC will be calculated per each client with consideration of averaged RHC per client categories and the difference between its price factors to client related price factors, i.e. system complexity, etc. FIG. 16 shows a graph 290 of periodical risk-hedged contract price corrections of a method in accordance with an embodiment of the invention. RHC are set to the client's subscription package(s) period. RHC are set with cap on services value to avoid client abuse of this method; equally, RHC price are set to individual client with calculation of profit margin variable; which is correlated to overall profit margin

calculation, this correlation is calculated per average price of client categories. RHC price is agreed to the period of subscription (e.g. of one year); but, RHC price is updated periodically (e.g. half/quarter/monthly year) in order to: reduced risk of loss of profitability to Grid's parties and to encourage RHC economical usage by clients; hence on select period any volatility between expected price (average RHC category+ profit margin) and real price (re-calculate average RHC category + profit margin) is reduced by setting new average price for the next period, re FIG. 16 – Periodical RHC Price Corrections; this RHC price correction are capped to certain price; this is to maintain risk spreading between clients while maintaining client encouragement to uses RHC economically. Also, the risk covers by RHC are required to be with high degree of independency, avoiding impact by few risks across many clients; RHC with subscription are devised to increase efficiency by use of economy of scale (i.e. low performance monitoring and analysis total cost) with flexibility of service as per system operation during its life cycle. Also, once the business model reaches critical mass stage then expected price reduction are encouraging increases of client base with direct impact of increase RHC volume; high correlation between subscription and RHC volumes is maintained due to setting fixed overall profit margin on the top of average RHC prices per client categories. RHC are devised as the core gross profit margin for all Grid's parties across all services provided to clients, beyond subscription.

Therefore the more clients use this business method the more efficiency it bring to end client in term of cost offering and more profit to the Grid's platform and service providers parties. This is until it reaches equilibrium state of efficiency vs. profit as per target markets. Further Grid related efficiency and profit increases is by expanding service offering on the Grid's platform, re FIG. 15 – Subscription and RHC Model. FIG. 15 shows

a graph 280 of subscription and risk-hedged contract of a method in accordance with an embodiment of the invention

A computing/communications grid framework 10 is proposed in accordance with an embodiment enabling the provision of performance analysis services by different and separate entities remotely, as discussed with reference to FIG. 8. The preferred embodiment of the system has three components: Analysts 14,16, Computing 12, and Collaboration 18 resources. All analysis service activity starts with on-going or on-demand service request (SR) by clients and ends with an analysis report dispatched electronically to clients. This will happen within an agreed period determined by client's subscription package (on-going SR) or by a specific quote (on-demand SR) from the main service provider. Service may be subscription, pay-on-demand, or risk hedged contract based. Subscription will offer subscription package based service to client, which is the package that is most appropriate per the expected pattern of minimum to average usage by client. Analysts' resources are divided into two groups according to activity type: Interactive analysis 14 with clients and background analysis 16. Interactive analysis is performed by service provider(s) staff together with clients' staff in performance projects locally. Background analysis is an on-going analysis by exception (forecasted or raised) and any other analysis that can be remotely performed. It can be provided by a main service provider or by contracted and specialized 3rd party service providers 19. Computing centers will provide the computing platform for processing clients' raw measurement-metrics and model data. This will be done by scheduling monitoring and modeling algorithms or by any other suitable specialized tools. Computing Centers locations will be strategically positioned to cover the relevant geographic area, usually worldwide. Data will be directed to the nearest center for processing. It will then be reviewed by the center's automatic monitoring process in order to immediately track or forecast abnormalities in normal workload patterns. When abnormality is detected it is raised and flagged out.

Background analysts will then conduct analysis or dispatch it to different service providers' teams. At any point in time clients are able to raise a service request via a Collaboration system (e.g. (1) Retail company has a big promotion campaign planned for six months ahead, raising concern whether company's mission critical system can cope with expected boost in daily orders during campaign period (2) Systems consolidation is planned for the next budget year where the company needs to identify an optimal solution and quantify its estimated cost, etc). Service requests are processed by service provider analysts resources. Analysts teams will interact with the Computing Center via the Collaboration system as needed in order to process modeling or other algorithms per clients SR. An analyst team may be locally allocated to client and analysts elsewhere. These team members will collaborate via the Collaboration system (re Grid Protocol).

Monitoring and modeling data can be fed back into the system's knowledge-base. This will enhance performance analysts' knowledge with regards to best practice concerning performance issues. In addition, it can be used to provide guidelines to system architects, and sold as performance-analysis content.

FIG. 4 shows a time-line of steps 40 of a method in accordance with an embodiment of the invention. Algorithm –Service Request Manager: The Performance - Modeling process is initiated by the Client / Analyst Web-Browser and completed when a report is submitted to the Client / Analyst respectively. All service requests are Report related service-requests. Such reports contain an analysis of model & data of Enterprise Infrastructure or specific Applications. All reports are compiled by Solution Provider Analyst and based on analysis (or issues escalation) completed on Grid platform. Any communication between Client / Analyst to SP's Analyst/System during logging SRs is performed via Grid Protocols (re Grid's Service Request Manager component).

Stages Description

A—Log - In

All Clients / Analysts can access the Apparatus only if their details are already registered successfully in with Grid's Collaboration Apparatus (GCA). The Client can access GCA's Client's Intranet by giving its User-Name and Password. Such details are listed in the Client-Profile details in the GCA designated CC's data warehouse. Once Client submit their details then a security check is initiated which will result in providing Client to access their Client area in secure environment or exception trigger. An exception trigger will result in access being rejected. Rejecting or providing access is recorded in a Client-Access log.

B—Modeling - Based Reports Selection

Client can make selection of a report from a list of reports available in the tagged area of Modeling-Based Reports. The Modeling-Based Reports include a pre-defined reports selection (re Report Templates below). Such report offering is based on availability of the Modeling-services on this platform. This reports list is updated on fly by periodically checks of the Web-Services Registry (re Collaboration Apparatus description). The reports under consideration at the point of compiling this requirement document are:

IT-Investment Justification :

Business Demand impact on Infrastructure (or Application) Analysis, Major Change Management Justification, Etc. Once Client makes a selection of report template then he/she can move to the next stage.

C—Client Details

Present screen with Client details such as: Name, Title, Contact Tel Number(s), Address, etc. Client asked if these details are still valid, or Client would like to update these details. If update is rejected then the Account Profile Management module is being initiated, re Web Services platform. Note that only Clients that are registered as a Client Area

administrator can update Client Profile details. Once Client confirms that he/she is happy with these details then he/she can move to the next stage.

D—Electronic - Billing Confirmation

Customers are billed in advance. The bill is initially based on an estimated quote. The next bill will follow on completion of the work if more was required then the total days quoted; then another bill will be submitted for the extra work required (any extended work is subjected to confirmation from Client on T&M basis). Clients are divided into three groups: subscribers, insured & Pay-As-You-Go. Subscribers have credit of number of work units per some services. This would be a variable sum according to the package the Client has selected. Pay-on-demand has no credit at all. Solution-Provider's Business Terms dictate that no work will be carried by Solution-Provider's Analyst unless it can be paid by Client's Outstanding Credit, or paid electronically to Solution-Provider bank account, re Solution Provider's Business Terms Document.

Present screen with Client billing details (bank account, etc.) Asked if these details are still valid, or Client would like to update these details. If update then initiated Account Profile Management module. Once Client confirms that he/she is happy with these details then move to the next stage.

E—Initial Consulting - Quote

All ad-hoc based Report selected cannot be estimated in advance. In these cases estimation will be carried over several stages. First Stage: Automatic quote is initiated by the Apparatus based on Report selected (e.g. For an IT-Investment-Justification Report a 500 work-unit is quoted (five full man-days of two staff for 2.5 days consulting). Second Stage: This stage is required in cases where it appears that the initial consulting period is not sufficient for preparing an estimated quote; hence, an extension on the initial consulting is required. Such extension requires Client confirmation once extension days are published. Third Stage (or second, if no extension on initial consultation is required):

After initial consultation the Consultant publish a quote on the Apparatus). Fourth Stage: It may be that the estimated work isn't sufficient to cover the Report analysis. If this happens an extended quote will be published by the Consultant, to cover the rest of the work. It should be Noted that it is not expected to have more then one extension on initial consultation, as well as in respect with the extension towards the completion of the Report analysis. Should such cases arouse then Solution-Provider's Account Manager will deal with each situation on its merit. Details: Present on the screen initial quote estimated for Client to confirm. Once Client confirmed by accepting the quote (by hitting the Next button) then move to the next stage.

F—Confirmation

At this stage the Apparatus will log a Service Request and advise the Client on the Conditional-Service-Request-ID. Conditional-Service-Request-ID is a unique ID number associated with a specific Service Request selected by Client. This ID is conditional until funds estimated are allocated. On allocation a Service-Request-ID replaces the Conditional-Service-Request-ID, re Credit Check Step. The Service Request will stay open up to 72 hours from the time a Conditional-Service-Request-ID was submitted. Once funds are clear work can commence. Outstanding Service Requests without allocated founds for over 48 hours will be trigger automatically exception, which will route (via Grid's exception protocols) to Consulting Administration Staff (SP's Analyst with skill set of administration) of a problem then liaise with the associate Account Manger or directly with the Client, re Grid's Service Request Manager). Present to Client the information: outstanding credit left, the amount quoted per open service - request, the balance of credit/debit on Client's account. Asked Client to confirm the Service Request. Provides Client with Conditional-Service-Request-ID and advice on the following steps and expected timetable as well as on Customer Care contact number and e-mail.

G—Credit Check

Once a Client confirms the quote it will be informed on his/her outstanding left if any. If no sufficient credit is left then credit check is initiated. Should credit check fail, inform Client by e-mail (via exception protocols as SP's Systems with skill set customer service exchange) as well as inform Consultancy Administration Staff (via exception protocols as SP's Analyst with skill set of administration). They will liaise with Client or advise Client's Account Manager of the situation. Once Credit check is successful allocate funds: draw sum from any credit left, or transfer funds from Client Bank Account to Solution-Provider. Once funds transfer confirmed update Client records, update Solution Provider's financial records, generate a Service-Request-ID, and advise Consultancy Administration Staff (via exception protocols as SP's Systems with skill set customer service exchange) on initiation of Service-Request. Initiate a Credit Check on funds. Once Credit-Check passed initiate funds transfer. Once received confirmation on funds transfer: change Service request status from conditional to confirmed and generate a service request number to the Client. Advice Consultancy Admin. Staff on initiation of service request.

H—Consultancy Admin

Once a service request is confirmed then Consultancy resource(s) will be allocated to the estimated initial consultancy period. Such work will be scheduled according to availability of resource(s) and service terms (Client's Commercial Terms). Once Consultancy Admin. receives Service-Request initiation then: Allocate a Analyst from Analyst pool for the period quoted at the last estimation, then, Liaise with Client directly or via Account Manager to agree the starting day of Consultancy. Once determined update Service Request record on Web Services platform. Generate confirmation by Grid's protocol to Client on Initial Consultancy with details of the person who confirmed the Consultancy, (Note all conversation of Solution Provider's staff with Client are recorded. This is in effort to sort out any discrepancy in interpreting the communication between Solution Provider staff and Clients). Once confirmation is sent to Client a Consultancy-Work

Tracker is activated to monitor that initial consultation is completed per schedule. This is done by checking that Estimated-Quote was published at Client Area and Service Request has changed status. Any delay in publishing data will raised exception to Consultancy Admin. Staff and alert Account Manager via Grid's exception protocols.

K—Quote Preparation

Upon completion of initial consultation the Consultant(s) publishes estimation of work based Service Request; or initiating an extension for the initial consultation. The quote estimation is based on template (customized to the selected Report type). Once quote is published then Client will be automatically advised as well as Consultancy Admin all in effort to schedule main consultancy work. Once initial (including any extension) consultation is completed: Compile quote estimation using a template, Publish Quote at Client-Area, Once Quote is published an automatic notification via e-mail will advise Consultancy Admin. Staff to liaise with customer concerning the quote.

L—Consultancy Admin

Once a Service Request is confirmed then a Consultancy resource is allocated and actual Consultancy Work is scheduled. When Consultancy Admin. Staff receives Service-Request initiation then: Allocate Analyst from pool of Analyst for the period quoted at last estimation, re Consultancy - Resources Manager. Liaise with Client directly or via Account Manager on the start day of Consultancy. Once agree on the date then update Service Request record in Web Services platform. Generate confirmation by Grid's exception protocols to Client on Initial Consultancy with details of the person who confirmed the Consultancy~~,~~—Once confirmation is sent to Client then Consultancy-Work_Tracker will be automatically activated in order to monitor that actual-consultancy is completed as per schedule. Report-Published would be published at Client Area and Service Request would change status. On any delay in publishing data then an exception to

Consultancy Admin. Staff will be raised and Account Manager to be alerted Grid's exception protocols.

M—Report Submission

When Actual Consultancy work is completed then report is published in Client Area. If consultancy work needs to be extended then Managing Consultant will advise Consultancy Admin. Staffs who will update Service Request status to REVISED-QUOTE, then Step K & M need to repeat. In this case Consultancy Resources will continue work upon Service-Request job until Account - Manager will advise otherwise. Once Service Request job is completed: Managing Consultant will publish a Report (after Consultancy QA has completed) in Client Area. This generates automatically a notification for Client on Report availability; as well notify Client's Account Manager; if an Extension to consultancy work have been raised then Consultancy Admin. Staff / Account Manager will repeat Stage K & M for this extension; Account Manager to close the service request after liaising with Client. When Service Request hasn't closed 48 after the report is published then a notification will be generated to Consultancy Admin. Staff and Account Manager. This is on daily basis in the exception report.

Detailed Example

Solution Platform (Re FIG. 3)

Solution Providers (SPs) services worldwide. Service availability will be based on region-ability (E.g. Western Europe). FIG. 3 shows a table 20 listing a system of solution system platforms in a single integrated solution across three layers in accordance with an embodiment of the invention. The solution system platforms are tied together to one integrated solution across three layers, i.e. computing center 12, collaboration services 18, and system layer 10. The platforms shown in this embodiment are support platform 22, performance monitoring 24, consulting platform 26, performance modeling 28, data warehouse 30, reporting platform 32, data mining 34, and production system management

36. It will be appreciated that other embodiments may comprise of different platforms and arrangements. Solution Provider's platforms are divided to the following categories:

- ***Collaboration Apparatus 18*** - The platform manages a cycle, which starts with Clients' Service Requests (SRs), Analysis or Support SRs, and ends with SR results being dispatched to Clients, re FIG. 5. Development environment is BEA-IDE. Scalability is based on clustering technology (e.g. WebLogic by BEA Systems, Inc. of San Jose, California, United States of America), where identical servers are aggregated as a cluster. SRs are initiated from Clients or automatically by Apparatus performance monitoring platform or by SPs. The Collaboration platform is using for these entities a connection over Internet via Web-Services protocols and directories (e.g. SOAP, etc). The Collaboration data model deals with Clients (subscriber or 3rd party solution provider based) profiling and package details, billing details, Solution Provider (SPs) resource allocation, SRs, etc. Clients can interface with SPs systems automatically via Web-Services protocols, once the commercial basis has been established. Embodiment implementation involves the following steps: Client's Software (S/W) lookup at Web registers and set-up the links to Solution Provider's Web-Services; links will be established automatically by using Java 2 Platform, Enterprise Edition (J2EE) by Sun Microsystems, Inc. of Santa Clara, California, United States of America, Microsoft.NET by Microsoft Corporation, Redmond, Washington, United States of America deployment. The main algorithms performed in a collaboration apparatus Grid-Protocol: SPs interaction management as per technical analysis process; (2) load balancing optimization of the computing center's job as per SLA contract term of SPs with Clients; Collaboration system is a 4 tier web-server and application server farm (HP 5670 by Hewlett-Packard Company (HP) of Palo Alto, California, United States of America base with Red-Hat Linux operation system by Red Hat, Inc. of Raleigh, North Carolina) that enables connectivity between entities

participating in this process: Clients, 3rd party who manages Clients systems and service provider (SPs)

- ***Computing Centre Apparatus*** - (re **FIG. 2**): Computing centers were designed as 4 tiers web-servers and application-servers farms (HP 5670 with Red-Hat Linux) with scalable technology. This is an integrated solution of several Apparatus' platforms. The Computing Centre is where all calculation of statistical analysis or mathematical modeling algorithms is performed. Its scalability is based on clustering technology (e.g. BEA's WebLogic), where identical servers are aggregated as a cluster. The main tasks performed in a computing centre are:

(-) Data Warehouse: (1) Loading Enterprise-Infrastructure/Application(s) measurements metrics data, architecture model data, system / transitions logs or system transaction traced data from Clients sites; (2) Data Mining and Normalization to format raw measurement data; (3) Data storage for algorithm's data results.

(-) Algorithms: (1) Monitoring against SLAs/QoSs/PoFs targets/values; (2) generating PoFs; it is an iterative What-If analysis of PoFs' associated models (synthetic workload model based).

Performance Monitoring Platform (Re FIG. 6)

Monitoring Against SLA / QoS / PoFs targets

Allows visual monitoring of subject-matter system' architecture deployment model (SDM) , based UML script; and setting performance and availability level targets, where possible; using graphical – aid. Offers to have a pre-defined SLA and QoS targets for Clients as a starting point. When targets are reached then Exception Handling is initiated, re Operation Exceptions Management below. ***Interactivity***: asynchronous near real time calculation. ***Data***: database's table. ***UI***: For each Worst Case Scenario Model (WCS) there is at least

one SDM; SDM can be visible by updating its content per each active SSM model or SDM is visual for each WCS; this is defined in the system's parameters, e.g. one SDM with record visualized per current active SSM or six SDM are visual per six WCS model; Setting SLA and QoS targets using interface to Enterprise-Infrastructure/Application(s) Architecture Mapper method with capability to set each object's record (e.g. Browser to Application Server response time = 0.01 sec, Database Server CPU Utilization set to 90%) , re FIG. 7. ***Algorithm***: A suggestive pre-defined SLA/QoS/PoFs are calculated and appeared automatically; initially calculated as current mean value of performance metrics plus 2-3 sigma

SD (define in Apparatus parameter). Client / Analyst can modify these pre-define SLA/QoS using the Enterprise Architecture Mapper interactive tool method (based UML scripting) by setting the record field using generic object' method, e.g. performance metrics can be set (latency, throughput, path length, etc). PoFs are visually and clearly indicated, if SPM have been created and simulated with positive PoFs identification; they visually appear at all times using small screen window visually linked to the subject-matter system's architecture resource in the SDM; ***Business Logic:*** J2EE/.NET component. This component is wrapped around the Enterprise Architecture Mapper component; as the QoS/SLA/PoFs component is required the Enterprise/Application Architecture Model is to be used as a base platform for setting up the SLA/QoS/PoFs level.

Forecasting

Forecasting includes linear, non-linear, exponential and s-curve projections. Data is saved for 3 months by defaults for an historical trend analysis. Forecasting is used for calibration of PoFs results and resource capacity measurement for some system resources such as memory utilization; forecasting calculation is by default to all PoFs related systems performance parameters; it is calculated in intervals of days of up to 1 months forecasting

data. It is used in conjunction with PoFs analysis for short term issue exceptions. ***Interactivity:*** Batch job by default. The business logic will be initiated by mostly scheduled (batch) jobs or ad-hoc (by setting default 24 hours re-calculation, set as trigger in database) ***Data:*** High. As a result of data transmission to and from the Automatic Forecasting Server (e.g. SAS High Performance Forecasting) product need to have high bandwidth (fiber optic / Symmetric by EMC Corporation, Hopkinton, Massachusetts, United States of America). Usage of Data-Warehouse/Data-Mining product for calculating in advance and near real-time of the algorithms parameter sets. ***UI:*** forecasting data is loaded by Monitoring against SLAs / QoSs Targets component by updating the forecasting databases table, minimum every 24 hours. ***Business Logic:*** J2EE/.NET component, script programming language to interface the Forecasting server. ***Interfaces:*** Automatic Forecasting Server, Statistical Server, Data Warehouse Platform ***Complexity:*** High ***Templates:*** End-to-End response times, CPU utilization/ memory, Disk, Network analysis, etc.

Operation Exceptions Management

Exception Management for: (1) PoFs exception handling, with 2-3 sigma SD error windows (defined in Apparatus parameters); (2) Setting SLA/QoS level values as per subject-matter system's architecture resources and performance requirements; (3) performance analysis requests by SPs' Analyst/Systems. In case of SLA/QoSs/PoFs the exception is triggered on SLAs / QoSs / PoFs values level, set by default or by Analyst in SDMs models; it is an database trigger function based; re Monitoring against SLAs / QoSs / PoFs Targets component. Exception is processed in sequence between SPs until resolved, or part of exceptions' pattern process between SPs until resolved. Exception records are inserted / communicated by initiating a call to Grid's Technical Analysis Protocol – Management component method. ***Interactivity:*** High ***Data:*** Low ***UI:*** Interface to

Enterprise/Application Architecture Mapper. ***Business Logic:*** J2EE/.NET components, SNMP API for action/broadcasting exception message to SPs' Analysts (by skill-set)/Clients/3rd Party's enterprise management systems or e-mail the same respectively ***Interfaces:*** *Grid Technical Analysis Protocol – Management.*

Enterprise/Application Architecture Mapper

Automatic generation of subject-matter system's architecture models from the following sources: (1) model generated by network/enterprise management tools (e.g. IBM's Netview) when converting their output model in XML format to SDM (based UML); thereafter calibrated this model manually whenever it required; (2) automatic generation of Application Architecture from model generated by tools such as Profilers, Agent based model auto discovery; (3) UML design case tools; etc. All SSMs are linked to their SDMs, re PoFs Projection Manager component. ***Interactivity***: High. ***Data:*** High. Need to have high bandwidth (e.g. fiber optic / EMC - symmetric) connection, usage of Data warehouse product (e.g. Teradata by NCR of Dayton, Ohio, United States of America). ***UI:*** Visual Model Architecture view is SDMs (based UML deployment Diagrams) and SSMs (based UML sequence Diagrams) based. With a palette to add / takes nodes. All Nodes and Arcs are described in Modeling Language standard (UML or similar). All performance metrics appear in the system deployment DIAGRAM view: For the above screen a lookup screen view (initiated by mouse - double click) with details on these Nodes (e.g. component ID); SLA/QoS any other system resource forecasting, excluding PoFs, will be visual in window, where activation is by clicking the option set mouse right button; PoFs prediction will appear in separate window with the capability to save/delete baselines if what-if analysis is conducted, it will be visual in separate window, where activation is by clicking the option set mouse right button; per individual system resources: SDM visual mapping with distinguishes connection point of path between nodes of architecture of response time metrics data (metrics set by Performance Analysis Measurement component, re below).

Business Logic: J2EE/.NET components. ***Interfaces:*** This component is warping around Performance Analysis Measurement component (re below) and Forecasting component (re above); as all performance metrics data is required to be calculated per all Model's Nodes.

Performance Analysis Measurement

Measurements of end-to-end performance metrics Performances Metrics cover are: Latency, Path Length. For example for those Clients with cross platform workflow a compass Unix (UNIX is a registered trademark in the United States and other countries of the Open Group) and Windows by Microsoft will allow viewing the correlation of performance info within and across systems revealing dependencies between performance metrics. It's necessary for diagnosing response time problems for system span platforms. Batch job by default, but may be initiated by ad-hoc calls. **Interactivity:** High. **Data:** High. Need to have high bandwidth (e.g. fiber optic / EMC - symmetric) and use of Data warehouse product for data manipulation. **Business Logic**: J2EE/.NET component. **Interfaces**: 3rd party components and/or use of transaction log data output from 3rd party component based ARM 2.0 standard - transaction tracking functions.

Resource Share Management

This data is essential to map between performance trend (PoFs, short term forecasting) and system resources capacity. Allowing to have various snapshoot of system-resources performance (E.g. CPU: Queue-Length, etc). **Interactivity**: Batch by default. **Data:** Low **UI**: None **Business Logic**: J2EE/.NET components. **Interfaces**: **Algorithm:** for every subject-matter system performance parameter check the difference between PoFs value to short term forecasting (re Forecasting component) if value above/below 1-2 sigma SD of PoFs then exception handling is generate, re Operation Exceptions Management.

63

(a) Monitoring-Execution Scheduler

(1) Invoke Grid's Optimizing Resource Manager component via Web Services protocol with scheduling parameters; (2) Automatically scheduled performance monitoring algorithms run will be by subscription contract constraints and service priority. It allows manual updating of prepared schedules on exception requirements. Provides easy interface to be accessed anywhere on Solution Provider's platform. Dynamic estimation of jobs completion schedules. Dynamic view of work progress schedule. Dynamic triggering of alerts when scheduled jobs fail to complete. Allow booking time-slot for monitoring algorithm run. **Interactivity:** Medium. **Data:** Low. **UI:** J2EE/.NET component. Business Logic: J2EE/.NET components. **Interfaces:** Grid's Optimizing Resource Manager component via Web Services protocol (re below).

Performance Modeling Platform (Re FIG. 7)

Service Request Manager

The Performance - Modeling process is initiated by the Client's/Analyst's Web-Browser and completed when a report is submitted to the Client/Analyst respectively. All SRs are report related service-requests. Such reports contain model analysis with description of SDMs and their SSMs with detailed description assisting drill down fault diagnostic follow-up or technical decision; or details associated with escalation of issues by SP's Analyst/System (re Exception protocol - Problem Escalation ID). All reports are compiled by SPs' Analysts/Systems and based on: (1) analysis completed as per Clients original/modified models (SSMs and SDMs based), generated via Performance Modeling

platform; (2) analysis completed by SPs own systems, where their analysis is with respect to Client's SSMs and SDMs; SRs related records are inserted / communicated by initiating a call to Grid's Technical Analysis Protocol – Management component method.
Interactivity: *High* ***Data:*** *Low.* ***UI:*** *of 3rd party package used and few screens for routing logging SRs via Grid's exception protocol* ***Business Logic:*** *of 3rd party package used (reporting tool: Business Object and E-Commerce tool).*

Modeling Editor
Model Editor can do all functionality of Enterprise / Application Architecture Mapper (re Performance Monitoring). It enables modeling at any system's stages during its life-cycle. Provides Model Design Patterns, in an effort to reduce the time required to build the model. Models types are linked (.e.g. SSMs models to SDMs). This tackles the need for analysis as per complexity of layer of specification on top of the current layer to tackle a more specific modeling problem domain. If Enterprise Architecture Mapper component provides the SSMs and their SDMs model with PoFs, then the Modeling Editor allows: (1) generating models for systems without trace data; (2) enhanced/modified subject-matter system's architecture design, now that a fully calibrated performance model is available. Models are generally built using drag-and-drop graphical representations of pre-defined building blocks as per model type (e.g. UML Deployment diagram, UML Sequence Diagrams, etc), such as queues, servers, transport entities, and logical modules (e.g. if-statements and while-loops). Model building blocks ('design pattern') can be modified into customized building blocks, as required, per model's problem definition. Such modification relates to an industry related component domain. ***Data:*** Medium. Models objects are stored in several tables. ***UI:*** Swing or Java Server Pages (JSP) by Sun Microsystems, Inc. /Servlets. ***Business Logic:*** *Java* components by Sun Microsystems,

Inc.. ***Interfaces:*** This component is wrapped around Enterprise Architecture Mapper component.

Modeling Workshop Manager

Manages all modeling from one environment by pre-defined methodology selection. Covers initiation of all modeling processes, from model creation up to end-result by dispatching a Report to Client-Area. Allows managing several models object concurrently, e.g. one can perform performance modeling of the same model created/updated using Modeling Editor to conduct performance modeling using analytical modeling of choice and simulation modeling of choice – Discrete Event Simulation, concurrently; it hence allows to calibrate a analytical performance model; which will, at the end of this process, run to provide PoFs prediction, etc; Model object fields are: model ID, state positions, etc; call to components interfaces gives model object by reference. The methodology's routing is hard coded in the component's methods from design simplicity reason. The component main function is to manage the interactivity routing needed at formulating subject-matter system's architecture models and at modeling stage. Such design allows update modeling methodology or interfaces from one component. ***Interactivity:*** High Data: Low ***UI:*** Swing or JSP/Servlets ***Business Logic:*** Java components. ***Interfaces:*** to *most performance modeling components, Reporting Server component*

Modeling-Problem Formulation Wizard

Wizard tool approach with reference to underlined type of problem (the type of Report required to produce by template generated per report selection). The wizard, a rule based tool enables the use of best-practice by assisting Analyst to formulate the following: (1) Identify problem domain based best practice rules; rules definition using pre-defined templates; (2) Formulating the problem definition (processes and data aspects). Wizard

offers a selection of problem patterns that can speed up the process of modeling - problem formulation. Pre-defined templates are hard coded using script language (e.g. extensible markup language (XML)). These templates are loaded at request during the Wizard process where selection is made on tables link to load and make a selection on another table, etc. On selection a call is made to a component (using object invocation) and component's objects fields are set with data selected during the Wizard process. Upon selection made on templates, during the process, a model data object is extracted from the model based UML (every general model is made of linked models, i.e. SSM with SDM and SSTM) to SPM; extraction from UML is made in the Methods Selector component, which calls interactively to set a performance model based selection to be made in the templates. It is the objective using this Wizard, to have a fully configured SPM with specific modeling method selection, e.g. SPM based Stochastic Process Algebra (SPA). ***Interactivity:*** High ***Data:*** Low. Use of templates (tables stored in database) ***UI:*** Swing or JSP/Servlets ***Business Logic:*** Java components. ***Interfaces: Modeling Workshop Manager Component.***

Methods Selector

Select a method of choice from a list of methods. Use functionality offering of methods libraries, re Simulation-Modeling Methods Management bean, Analytical-Modeling Methods Management bean, Numerical-Modeling Methods Management bean. Provide guidelines to assist in the selection of a method, re Modeling-Problem Formulation Wizard. This component functions as an intermediary to configure SPM based selection which is made in a Modeling Problem Formulation Wizard component; such selection is made on a model object passed by Modeling Workshop Manager component. The model simulation's selected method is set as a reference in Model object, e.g. Stochastic Petri Net (SPN); actual extraction from UML to performance model is based on existing techniques that is initiated when a call is made to basic functions of simulation libraries. Such

techniques are discussed in Hopkins et al., "Two Approaches to integrating UML and Performance Models", Workshop on Software and Performance, Proceedings of the third international workshop on Software and performance Rome, Italy, Pages: 91 - 92 , ACM, 2002, and Whittle et al. "Generating Statechart Designs from Scenarios" NASA Ames Research, USA, International Conference on Software Engineering, ACM, 2000. *Interactivity: None Data: Low UI: None.* ***Business Logic:*** Java components. ***Interfaces:*** Modeling Problem Formulation Wizard component, Simulation-Modeling Methods Management component, Analytical-Modeling Methods Management component, Numerical-Modeling Methods Management component.

Model Experiment Manager

Tool's point-and-click interactivity translates into faster answers and also quicker and easier re-stating of problems. For example one can modify copy models of subject-matter system's architecture used for performance monitoring facilities (SDMs and SSMs models) and will measure and animate the performance of the modified model by Simulation. The component functions as a What-If analysis object. Updates made to the model object (one model per SSM and SDM per) are automatically converted to unified models, based UML, into SPM per simulation method selected earlier using Modeling Workshop Manager; the animation capability is via simulation package's functions; on completion simulation run and then the simulated values (including PoFs) in the model object are updated; Saving the model object in the database is by using menu/mouse-click save selection. **Connectivity**. Initiated Report Wizard and save Report. Can also search for an optimal configuration of a system interactively by optimizing PoFs set. ***Interactivity:*** High ***Data:*** High ***UI:*** Swing JSP/Servlets ***Business Logic:*** Java components, wrapped around Modeling Editor component. ***Interfaces:***

Simulation-Modeling Methods Management

On component creation an extraction of SPM is made from it linked SSTM by used of PEPA upon SSTM's statechart and collaboration models result in PEPA model, which can used as input to: (1) Mobius software, which can generate distributed discrete-event simulation. Run simulation until a certain confidence level is statistically achieved. Initiate a simulation package libraries functions based Simulation Modeling upon performance model (i.e. model object) ***Interactivity:*** High ***Data:*** High ***UI:*** Swing JSP/Servlets ***Business Logic:*** Java components wrapping around Modeling Editor component.

Analytical-Modeling Methods Management

On component creation an extraction of SPM is made from it linked SSTM by used of existing techniques upon SSTM's statechart and collaboration models result in performance model (e.g. layer queuing network (LQN))

An approximation technique is used to model deterministic service times. Initiating a simulation package libraries functions based on Analytical Modeling upon performance model (i.e. model object) ***Interactivity:*** High ***Data:*** Medium **UI:** Swing, JSP/Servlets Business Logic: Java component. ***Business Logic:*** Java components wrapped around a Modeling Editor component.

Numerical-Modeling Methods Management

On component creation an extraction of SPM is made from it linked SSTM by: (I) used of PEPA (used with exponential probability distribution) upon SSTM's statechart and collaboration models result in PEPA model, which can used as input to: (1) PRISM 2.0 tool , which can generate the numerical models of: discrete-time Markov chains (DTMCs), continues-time Markov chains (CTMCs) and Markov decision processes (MDPs); (2) IPC

compiler which can generate DNAmaca format, used for analyzing very large Markov models; or use this model for Stochastic Process Algebra (SPA) modeling; (II) use of Stochastic Petri Net (SPN) existing software (e.g. Trivedi's SPNP) Markov chain and similar techniques. Initiate a simulation package libraries functions based Numerical Modeling upon performance model (i.e. model object) ***Interactivity:*** High ***Data:*** Medium **UI:** Swing, JSP/Servlets ***Business Logic:*** Java component. ***Business Logic:*** Java components wrapped around Modeling Editor component.

Models -Execution Scheduler

(1) Invoking Grid's Optimizing Resource Manager component via Web S*ervices protocol* with scheduling parameters; (2) Providing automatic scheduling of simulation and other models to run within specified period. Allows manual updating of prepared schedules. Provides easy interface to be accessible anywhere on Solution Provider platform. Dynamic estimation of jobs completion schedules. Dynamic view of work progress schedule. Dynamic triggering of alerts when scheduled job fail to complete. Allows to book a time-slot for a model run. ***Interactivity:*** *Medium Data: Low* ***UI:*** *Swing or JSP/Servlets* ***Business Logic:*** *Java components* ***Interfaces:*** *Grid's Optimizing Resource Manager component via Web Services protocol (re below).*

Data Warehouse Services Platform

Support all Solution Provider's platform by doing Data-Polling from Clients or 3rd Party sites. Data Validation (completeness, Integrity) Data Normalization: Mapping and Calculation of the parameters sets required by the algorithms required to run as per Client's subscription package (or 3rd party SPs contract). **Data Volumes:** It is estimated that data size requirements per Client / SP is 15-20mb. The number of Clients at any point in time is 250-500 per Computing Centre therefore data-volume requirement at any point in time is

about: 5 TB – 10 TB (tera-data) **Flexibility**: Medium. Need to maintain balance of developing generic DW function (e.g. Teradata bespoke API). **Performance**: High. Performance is the key consideration when selecting a DW package. Teradata was selected based on a benchmark result of TCP-H (year 2002) where its price/performance ratio is the highest in the category of DW with data above 1000GB. Availability: 7 X 24

Measurement Collection Management

Primary Requirements: FTP-based pulling from Clients' site, subject-matter system resources' measurement data files. Database-to-database transmission of subject-matter system resources' measurement data records. This is the preferred choice as transmission is quicker and reduces effort at the data warehouse point of saving the raw data at the database. Measurements data originating at files/records collected by the enterprise management system of choice at client/3dr party (e.g. HP-OpenView or IBM-NetView, etc). Data filtering: the selection of data needed to be loaded on platform's Solution (Computing Centers, Collaboration) server must be very rigid; only data required per client package is transferred to Solution platform's (Computing Centers, Collaboration)'s server. Data filtering is by using data scripts, which are client-package-templates based. Data Transfer: Central connection topology: Use of pulling servers, which are connected to NCR-Teradata via high speed cables (e.g. Fiber Optics, etc). The Pulling servers are set as a cluster, allowing scalability and failover consideration. Alternatively using Network connection topology: use of pulling servers, each located to serve a subset of Clients within a geographical area. The servers are all connected to a Computing centre via high speed cables (e.g. Fiber Optics, etc). Scalability is achieved by installing a new server per geographical location. Failover is achieved by setting a group of 2-3 servers, within close geographical proximity as a cluster. Use high speed transfer platform Validation on transfer success Use log files/database records Use of pulling package component. Check

for data size expected and that data details item are consistence with data head-footer if send. **Optional Requirement:** The performance Agent (wrapped around package component) is collecting system metrics from: App.-Server, Database Server, Web-Server, File/Print Server, PC Desktops, Unix Desktops and do: Time-Stamp / Log data in database. Performance Agent: Low overhead continues collection and logging of more than 300 global, process, application and system metrics. Discover the devices determining what data should be collected and how frequently Proactive performable reporting on widely used IT infrastructure building blocks including: Routers , Frame Relay , local area network (LAN), wide area network (WAN), remote network monitoring (RMON)/RMON2, and ATM Agents are available from system-management vendors and third party that literally support hundreds of platforms, including hardware servers, network devices, applications and database servers. Automatic data collection, with low overheads, typically < 2%, at very high granularity. Data-Collection scalability. By default, data is collected every 60 seconds, but one can easily change it to fit ones needs. ***State:*** State-full and Asynchronous Algorithms. Need to identify a method to make the data collection agent (on client's production platform) light as possible for the performance monitoring. ***Interactivity:*** Batch by default ***Data:*** Very High. Must use high bandwidth and compression techniques **UI**: ***Business Logic:*** Java components wrapped around 3rd party data pulling component. BEA-WebLogic Server – Clustering API.

Normalized Measurement Data

Comparison and correlation of application and system resource measurements. Control data aggregation and sophisticated analysis to support reports. Data aggregation into periods of time, such as 10 minutes, 1 hour, or 1 day. Data formulation to parameters set required for performance monitoring and modeling algorithms. ***Interactivity:*** *Batch by default* ***Data:*** *High.*

UI: None ***Business Logic:*** *Java components, 3rd party components (simulation, analytical modeling), NCR-Teradata API.*

Analyst - Resources Manager

Analysis resources are scheduled on projects and on-going assignments via this component. The scheduling is set automatically per criteria of: availability, skill, location, Analysis type (back-office or Local-office site), Analysis duration and link assignments. The scheduler can be updated manually by Analysis admin staff. General resources booking principals are applied to allow capacity for future booking, etc. Allowing analysis of: resources productivity, exception ratio per work booked, Analysis work growth by skill, etc. Such data can be with graphical interface (E.g. Microsoft's Excel). ***Interactivity:*** *Batch by default* ***Data:*** *Low.* ***UI:*** *None* ***Business Logic:*** *Java components, 3rd party components – scheduling package.*

PoFs Platform (Re FIG. 9 and 10)

Worst Case Scenarios (WCSs) Identification

Identify WCSs as per key system parameters. By: (1) Analyst make a selection upon pre-defined list (of key system/component performance parameters (KSPPs/KCPPs) list) by making selection with respect to subject-matter systems' architecture (or subject-matter system's component) type, e.g. in the case of distributed middleware component, then key middleware component's performance parameters are: CPU Queue, I/O rates, etc; a list is visualized by scroll-down window; the pre-defined KSPPs/KCPPs are in a template format; these templates are industry best-practice based; (2) Analyst set search constraints using a defined criteria window; constraints are: elapse time search (e.g. one hour), Period Range: start end period (e.g. 1-3 months), period (Monday-Friday or Monday-Sunday), Date Range (8-12pm or 12pm-12am), etc (3) a simple database search is initiated, store-

procedure base, of all WCSs per each selected KSPPs/KCPPs; (4) then a simple database search per all system/transaction logs for data with WCSs set (for all KSPPs/KCPPs) building intermediary data tables for SSMs model generating. ***Interactivity:*** *Medium* ***Data:*** *High* ***UI:*** *Few screens for making the selection at per WCSs search parameter* ***Business Logic:*** Java components. ***Interfaces:***

Worst Cases Scenarios Models Construction

Constructing all WCSs model based UML. By: (1) Construct SDM by the same method described in an Enterprise / Application Architecture Mapper component (re above); (2) for each WCS in WCSs set construct system scenario model (SSM) and system state model (SSTM) while linking them to SDM nodes , e.g. SSM entities iteration as per SDM entities; SSMs and SSTM models are based UML; SSTM is combine two objects model of: Statechart model based UML and Collaboration model based UML; the method comprises: (I) For each WCS, from WCSs set, create a new SSM and SSTM object; (II) Search through intermediary data tables constructed and for each WCSs set pointers per individual entity and conduct search at all logs, from start time range through its end; the search is for patterns of set event occurring in individual entities in the logs; a pattern is considered valid if there is a reference to a similar pattern possible in the SDM; hence the reference to SDM acts as validation rule; if a pattern is successfully checked or validity then the pattern is listed temporary in the linked events temporary tables; tables based XML to be converted to UML model script; (II) a check is conducted to remove a short live event and reduce WCSs to the events with the longest latency as per time range; a search is conducted in the intermediary pattern tables in order to match the rule of: pattern occur at 75-100% time in the time range (start $\leq$pattern time range $\leq$end range); pattern links not matching this rule are discarded; (IV) generate the selected SSMs and SSTMs (UML Statechart and UML Collaboration models based); SSMs model are convert from

linked events temporary tables; SSTMs are create by automatic generating their correlated SSMs using exiting techniques, and set measured values as per SSMs in the link table associated with their SDMs data; (V) at this point it may be that there are still too many WCSs for PoFs monitoring; Analyst can reduce such issue by iteratively: increase the upper limit of pattern occur (e.g. 90-100% occurrence in the time range), the 'real' minus the 'defined' in UI parameters or Analyst has a visual view of SDM with SSMs and their PoFs (re Modeling Editor) then reduce SSMs as see fit. On constructing SSMs then track and set the time interval between massages, along with the objects name; such probabilistic timings can use for discrete event simulation whenever occurred.

Interactivity: *None* ***Data:*** *High* (complete at computing center high speed data access hardware by NCR-Teradata data warehouse), ***UI:*** *Medium.* ***Business Logic:*** Java components. ***Interfaces:***

PoFs Identification

Using UPMF methods will produce System Performance Model (SPM) with PoFs. These PoFs are simulation based only, hence temporary values; without reference to computing requirements; it is hence required to calculate PoF for each current PoFt in FoFt set. Subject-matter System (or component) computing requirement is set by: (1) such data can be imported from Enterprise System Management (e.g. HP Open View, CA TNG); or (2) set manually in a table format – subject-matter Apparatus' parameters. Using model verification techniques can produce the probability, that in event of current PoFs the systems will reach the point of system computing jobs backlog level (CBBL); weighted parameters will be used to identify that some PoFs have more impact system failure then others; PoFs are set in model (based UML) only for probability above 85% of CBBL; the expectation is that Analyst, when initiating Model Editor component, will re-iterate WCSs models simulation using different period range until received 'positive' PoFs for all WCs.

Interactivity: *None* ***Data:*** *Low* ***UI:*** *None.* ***Business Logic:*** Java components, Model Verification package API,

Interfaces:.

PoFs Projection Manager

The component manages the PoFs process in line with the above logic. ***Interactivity:*** *None* ***Data:*** *None* ***UI:*** *None.* ***Business Logic:*** *Java components.* ***Interfaces:*** *PoFs Identification component, Worst Cases Scenarios Model Construction component.*

Grid's Protocol Platform (Re FIG. 12 AND 13A-B 4)

Grid's Optimizing Resource Manager

Automatically (or manually by designated SP) scheduled per monitoring and modelling task at Computing Centres. ***Interactivity:*** *High* ***Data:*** *Low* ***UI:*** *Few screens.* ***Business Logic:*** Java components. ***Interfaces:*** 3rd party scheduling package.

Grid Technical Analysis Protocol – Management

(1) When receiving problem massage then perform the following: load exception record if text then route for SP's Analyst attention; if formula then link to application method call invocation use standard protocol; (2) Analyst may act on the problem and generate a message with a success field set to 1; or the same but with adding a new problem (with new exception (set time limit and Problem Escalation ID – with is own SP's ID, SP's Analyst ID or designated SP's ID for type of problem, i.e. SP's ID (back-office call centre), SP's ID (Client's Account Manger)) and new problem inserts); or forward to the next SPs; SPs are selected at random from making a selection in a skill set list; (3) Initiate a new problem message; (4) Insert Exception record. ***Interactivity:*** *High* ***Data:*** *Low* ***UI:*** *Low.* ***Business Logic:*** Java components.

Interfaces:

Grid Technical Analysis Protocol – Exception Handling

Insert an exception record in Exception table, access via Grid (re FIG. 8). The record is set by: (1) Computing center (CC) ID , unique 3 bit code; (2) Exception ID, unique 17 bit code; both together set a unique exception primary key; it is excepted that all exceptions will be cleared within one month from the Grid; (3) Text / Formula ID, 1 bit code; 0 for Text, 1 for Formula; (4) Description ID , a 24 bit, if a Text/Formula is 0 then it gives reference for a text record to be used by an Analyst, if 1 then links to an XML script to be used by SPs application rather then an Analyst; text or Formula fields are set by Analyst; automatic exceptions generate text only error messages; (5) Time Limit value, date/time stamp + SLA/QoS/PoFs values; a value which SPs are contracted to handle exception by client package/contract, this value is defined in Apparatus' parameters as per client service package; this value is monitored by a database trigger who initiated a call for review of this exception by Analyst; database trigger is set in the exception table; when it activated it sent error massage to the Problem Escalation ID (re below) to be managed by designated SP's Analyst manually; (6) Problem Escalation ID, 14 bit code; this field is set according to the context of exception, which is SLA/QoS/PoFs based and ad-hoc technical analysis including some scenarios of logging SRs; few examples: in case of some of logging SRs it set to Client Account Manger ID (SP's ID with skill set equal account manager); or in case of PoFs then it set to Back-office Analyst (SP's ID with skill set equal back-office analyst for this CC), etc; ***Interactivity:*** *Low* ***Data:*** *Low* ***UI:*** *One screen with Formula/Text error insertion by an Analyst, initiate interactively on ad-hoc basis.* ***Business Logic:*** Java components. ***Interfaces:***

Grid Technical Analysis Protocol – Problem setting Handling

Inserting a problem record in Problem tables, access via Grid (re FIG. 12). It set: (1) Problem ID, 17 bit code; (2) Computing center (CC) ID, unique 5 bit code; together set a unique primary code; (3) skill set ID, 6 bit code; identify the type of technical analysis needed; this value is map per exception error origination; e.g. (I) PoFs exception on CPU/IO then to a Background analysis SPs; (II) SLA/QoS triggering a exception error as per component where few SPs have visibility, it is forward to one of them; (III) Analyst comes across an issue that would need a complex analysis involved specialized SPs – Network Diagnostic SP, BEA's WebLogic Diagnostic SP, Switch Diagnostic SP by Cisco Systems, Inc. of San Jose, California, United States of America; (III) Analyst may wish to refer analysis from those SPs (re (II)) to another SPs for his review; (3) Problem Originator ID, 14 bit code; it set to SPs unique code in the Grid, as in example (I) and (II); or it set to the SPs code of the forwarded SP's ID, as in example (III); (4) success/fail ID, 1 bit code; o fail by default; 1 for success (5) Exception ID code, re Grid Technical Analysis Protocol – Exception Handling; ***Interactivity:*** *None* ***Data:*** *Low* ***UI:*** *one screen with fields that cannot be set by default.* ***Business Logic:*** Java components. ***Interfaces:***

Alternative implementation

Embodiment implementation is the only economical implementation viable using current technologies and that can be implemented in line with the concept of this embodiment.

The need for performance monitoring with modeling system's architecture is a fundamental requirement as long as its architecture structure is not trivial. System performance can be analyzed at source or remotely. The computing requirements of monitoring and modeling algorithms at large aren't feasible at source. Therefore, any performance analysis at source is limited in scope which leads to a degrading value of

analysis. Scope is limited as computing performance-analysis algorithms at source result in counterproductive performance improvement as direct impact of their own computing requirements. Therefore the only viable option is remote analysis. The best implementation of the embodiment concept is the one that balances system resources (servers, data storage, etc) with worldwide geographically distributed locations. This is in the aim to provide performance monitoring with modeling services to most enterprises worldwide and at an affordable price. Other considerations in the design of embodiment's implementation are the use of reliable components that can be integrated well as well as following best practices system architecture design.

Alternative applications

The embodiment can be applied to any sector where performance analysis can be applied to customer's system measurement metrics in the case of monitoring and can be applied to customer's system data model and/or workload trace-data in the case of modeling. The design of embodiment implementation allows scaling up with respect to any system under analysis and service is expected to be available worldwide. The only modification expected if the embodiment is used outside the IT/Telecom sectors is detailed design of some of the modeling algorithms, where the high level system design will stay intact. An embodiment method can be applied to any sector where performance analysis services are required. Examples of other sectors apart of IT/Telecom sectors where the embodiment can be applied are: (1) Manufacturing; (2) Engineering (3) Etc.

It will be understood that the method and system described above provides advantages. The commercial impact of this innovation technology is: (1) enabling better positive impact of systems, supports business processes, by maximizing adaptability of systems following the business dynamic; hence, improving business productivity and agile-

ability;(2) convergence of software engineering technology and methods with infrastructure management tools and method; (3) once successfully implemented it has the potential to allow 'just in time and just right' design modification by software and/or hardware components, hence reducing over capacity expenditure; equally it will reduce cost associated with system failure as per under capacity issues; (4) transparency, efficiency and increased quality and completeness of technical analysis at a cost effective price; (5) enabling platform framework to be built on or integrate further technology in any area of technology analysis. It will be appreciated that specific embodiments of the invention are discussed for illustrative purposes, and various modifications may be made without departing from the scope of the invention as defined by the appended claims.

CLAIMS:

1. A method for performance monitoring and analysis of a system, comprising: constructing a synthetic workload model of the system, comprising

   selecting a parameter indicative of system performance;

   monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run;

   constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM;

   constructing a system state model (SSTM) associated with the SSM, the SSTM representative of the state and static positions of the SSM;

   extracting a system performance model (SPM) from the SSTM; and

   applying a predetermined set of system conditions to the SPM to model and thereafter monitoring a point of failure of the system performance.

2. The method of claim 1 further comprising calibrating the SPM in accordance with predetermined set of system conditions to fine tune the SPM.

3. The method of claim 1 or 2 further comprising compacting the values of WCS by selecting the longest latency event sequences.

4. The method of any preceding claim further comprising re-iterating each step of the method with another set of system conditions.

5. The method of any preceding claim further comprising a graphical user interface to visually and interactively aid a user for performance monitoring and analysis of the system.

6. The method of any preceding claim wherein the step of monitoring a point of failure of the system performance is determined on a service level agreement (SLA) and quality of service (QoS) constraints.

7. The method of any preceding claim further comprising monitoring a service level agreement (SLA) and quality of service (QoS) constraints.

8. The method of any preceding claim wherein the system analysed is of a dynamic or a fixed configuration.

9. The method of any preceding claim wherein the method may be performed at any point in the life cycle of the system analysed.

10. An apparatus for performance monitoring and analysis of a system, comprising: means for constructing a synthetic workload model of the system, comprising

means for selecting a parameter indicative of system performance;

means for monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run;

means for constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the

SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM;

means for constructing a system state model (SSTM) associated with the SSM, the SSTM representative of the state and static positions of the SSM;

means for extracting a system performance model (SPM) from the SSTM; and

means for applying a predetermined set of system conditions to the SPM to model and thereafter monitoring a point of failure of the system performance.

11. The apparatus of claim 10 further comprising means for re-iterating each step of the method with another set of system conditions.

12. The apparatus of either claim 10 or 11 further comprising means for providing a graphical user interface to visually and interactively aid a user for performance monitoring and analysis of the system.

13. A method for providing an interface for a user to construct a system performance model (SPM) from a system state model (SSTM) representative of a system for performance monitoring and analysis in a modelling language (ML) environment, comprising:

selecting a modelling method with a method selector for extracting the SPM; and extracting the SPM from the SSTM with a modelling editor, the SSTM representative of the state positions of the system for the modelling editor to use for extracting the SPM, extracting the SPM from the SSTM based on a selected modelling method.

14. The method of claim 13 further comprising calibrating the models with the modelling editor in accordance with monitoring data received by the system analysed to fine-tune the SPM.
15. The method of claim 13 further comprising calibrating the models with the modelling editor in accordance with extracting another SPM by selecting another modelling method to fine-tune the SPM.
16. The method of any of claims 13-15 wherein the modelling methods are simulation modelling, analytical modelling, and numerical modelling.
17. The method of any of claims 13-16 wherein the modelling methods are Stochastic Petri Net (SPN), Stochastic Process Algebra (SPA), Discrete Events Simulation (DES), Queuing Network Modeling (AM), and Markov Chain (MC).
18. The method of any of claims 13-17 wherein the system analysed is of a dynamic or a fixed configuration.
19. The method of any of claims 13-18 wherein the method may be performed at any point in the life cycle of the system analysed.
20. The method of any of claims 13-19 further comprising a graphical user interface to visually and interactively aid a user for allowing the user to select, update and delete the point of failure models set with the modelling editor.

21. An apparatus for providing an interface for a user to construct a system performance model (SPM) from a system state model (SSTM) representative of a system for performance monitoring and analysis in a modelling language (ML) environment, comprising:

a method selector for selecting a modelling method for extracting the SPM; and

a modelling editor for extracting the SPM from the SSTM, the SSTM representative of the state positions of the system for the modelling editor to use for

extracting the SPM, extracting the SPM from the SSTM based on a selected modelling method.

22. A method for performance monitoring and analysis of a client system in a network computing environment, comprising:

providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, ,

receiving at computing center a service request from the client system for performance analysis of the client system, the service request posted from the computing center to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, ,

a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

23. A method for performance monitoring and analysis of a client system in a network computing environment, comprising:

providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, ,

receiving at network interface a service request from the client system for performance analysis of the client system, the service request posted from the network interface to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, ,

a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and

adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

24. The method of claim 22 or 23 wherein the protocol further comprises an exception protocol having a portion representative of exception data.

25. The method of any of claims 22 -24 further comprising the step of balancing multiple service requests with an application service provider, interconnected in the grid environment with the computing center, service providers, and the client systems.

26. The method of any of claims 22-25 further comprising the step of logging the service request in the network interface upon returning an executed service request.

27. The method of any of claims 22-26 wherein the service requests are received and posted in an ad-hoc workflow of performance monitoring and analysis of the client system in an asynchronous environment.

28. The method of claims 22-27 wherein the service requests are received and posted in a sequential or pattern routing manner.

29. The method of claims 22-28 wherein multiple service requests are processed concurrently in a large scale computer grid environment.

30. The method of claims 22-29 wherein each service provider, computing center, and client system have a unique identification.

31. The method of claims 22-30 wherein the protocol has a portion identifying specific computing center identification.

32. The method of claims 22-31 wherein the protocol has a portion identifying originating service provider.

33. The method of claims 22-32 wherein the service provider electing to process the service request, refer the service request to another service provider, or partially process the service request.

34.. An apparatus for performance monitoring and analysis of a client system in a network computing environment, comprising:

means for providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system,

means for receiving at computing center a service request from the client system for performance analysis of the client system, the service request posted from the computing center to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

35. A method for performance monitoring and analysis of a system in a network environment, a client connected with an application service provider via a network the method comprising:

receiving from a client a service request for performance analysis of a system;

conducting system performance analysis of the system upon receipt of the service request, the analysis performed with a unified systems modelling process and system monitoring process;

preparing a report of the system performance analysis of the system to the client for a fee.

36. The method of claim 35 wherein the a unified systems modelling process and system monitoring process generates a system performance module (SPM) of the system analysis, and the method further comprises conducting system performance

analysis based on the stored SPM upon another service request posted from client for the same system.

37. The method of claim 35 or 36 wherein the network is the internet.

38. The method of any of claims 35 to 37 wherein the method is performed in one or more of a computing center or a service provider.

39. The method of any of claims 35 to 38 wherein the step of preparing the report of the system performance analysis further comprises interactive analysis and background analysis.

40. The method of any of claims 35 or 39 wherein the step of preparing the report based on service packaging the report on a system performance criteria.

41. The method of any of claims 35 or 40 wherein the system performance criteria is point of failure.

42. The method of any of claims 35 or 40, wherein the system performance criteria is diagnostic.

43. The method of any of claims 35 or 40 , wherein the system performance criteria is what if modelling.

44. The method of any of claims 35 or 40 further comprising the step of charging the client for a service request based on client accessing grid interface.

45. The method of any of claims 35 or 40 further comprising the step of charging the client for a service request based on client subscription.

46. The method of any of claims 35 or 40 further comprising the step of charging the client for a service request based on risk hedging contract.

47. A method for performance monitoring and analysis of a system, comprising: constructing a synthetic workload model of the system, comprising

selecting a parameter indicative of system performance;

monitoring the parameter and identifying at least one worst case scenario (WCS) for said parameter during a period of time the system is run;

constructing in a modelling language (ML) a system scenario model (SSM) and system deployment model (SDM) pair on the basis of the WCS for each WCS, the SSM representative of system flow during the period of time, and the SDM representative of system components for monitoring during the period of WCS time, and the SDM mapped directly from the SSM; and

applying a predetermined set of system conditions to a system performance model (SPM) to model and thereafter monitoring a point of failure of the system performance.

48. The method of claim 47 further comprising constructing a system state model (SSTM) associated with the SSM, the SSTM representative of the state and static positions of the SSM, and extracting the system performance model (SPM) from the SSTM.

49. An apparatus for performance monitoring and analysis of a client system in a network computing environment, comprising:

means for providing a plurality of service providers interconnected in a network computing environment with at least one computing center and the client system, ,

means for receiving at a network interface a service request from the client system for performance analysis of the client system, the service request posted from the network interface to the service providers with a protocol having a portion identifying a skill set required by a service provider to process the service request, a service provider receiving the posted service request and determining whether to at lest particularly process the request based on the skill set information and adjusting

a portion of the protocol identifying the status of the service request; and returning the adjusted service request.

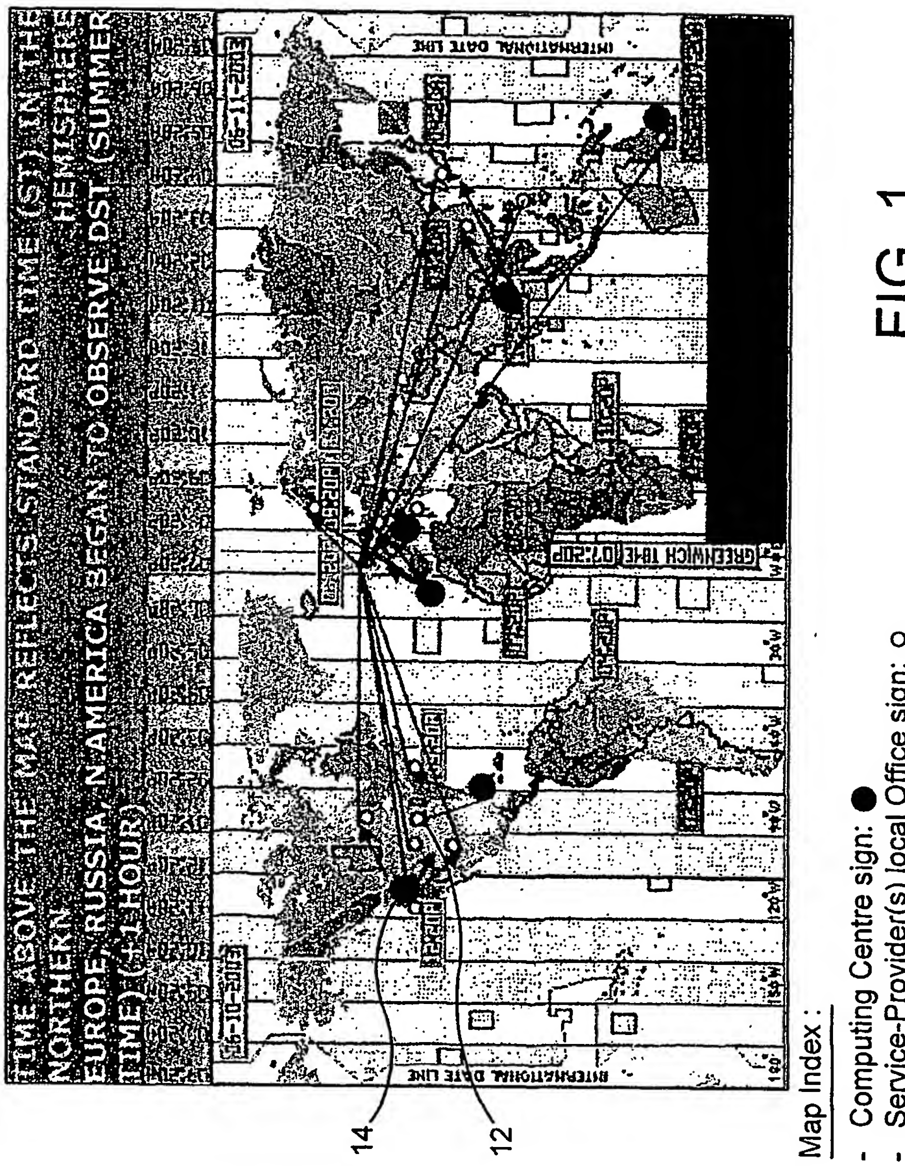

TIME ABOVE THE MAP REFLECTS STANDARD TIME (ST) IN THE NORTHERN HEMISPHERE
EUROPE, RUSSIA, N AMERICA BEGAN TO OBSERVE DST (SUMMER TIME) (+1 HOUR)
INTERNATIONAL DATE LINE
GREENWICH TIME 07:20P
14
12
Map Index :
- Computing Centre sign: ●
- Service-Provider(s) local Office sign: o

FIG. 1

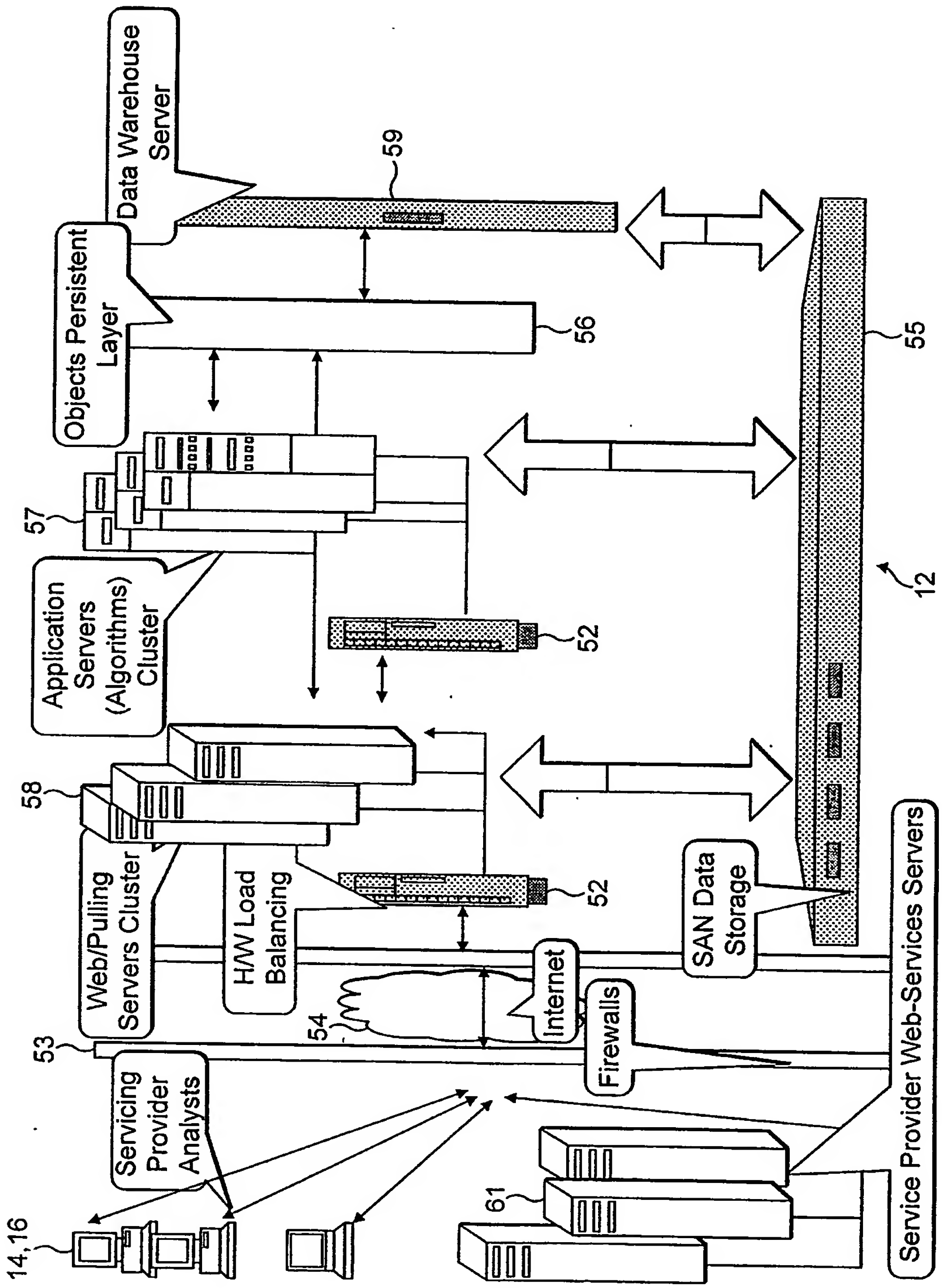

Data Warehouse Server
59
Objects Persistent Layer
56
57
Application Servers (Algorithms) Cluster
52
58
Web/Pulling Servers Cluster
H/W Load Balancing
52
55
12
SAN Data Storage
53
54
Internet
Firewalls
Servicing Provider Analysts
14,16
61
Service Provider Web-Services Servers

FIG. 2

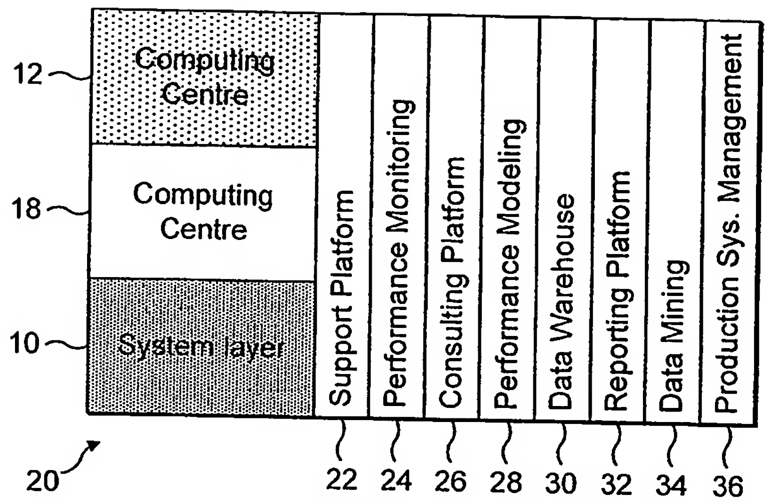

Computing Centre
Computing Centre
System layer
Support Platform
Performance Monitoring
Consulting Platform
Performance Modeling
Data Warehouse
Reporting Platform
Data Mining
Production Sys. Management
12
18
10
20
22 24 26 28 30 32 34 36

FIG. 3

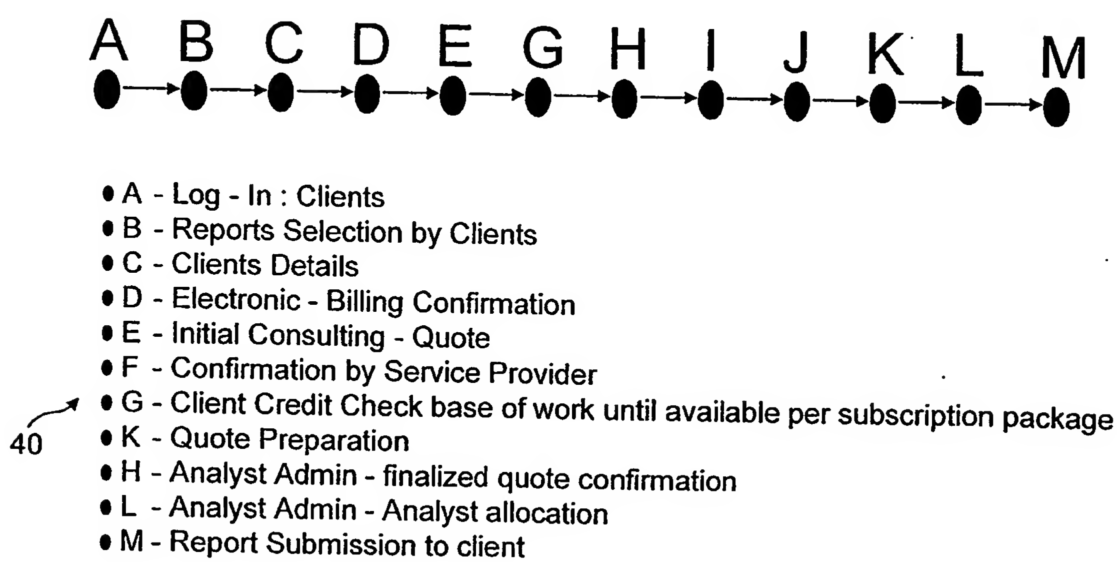

A B C D E G H I J K L M
40
A - Log - In : Clients
B - Reports Selection by Clients
C - Clients Details
D - Electronic - Billing Confirmation
E - Initial Consulting - Quote
F - Confirmation by Service Provider
G - Client Credit Check base of work until available per subscription package
K - Quote Preparation
H - Analyst Admin - finalized quote confirmation
L - Analyst Admin - Analyst allocation
M - Report Submission to client

FIG. 4

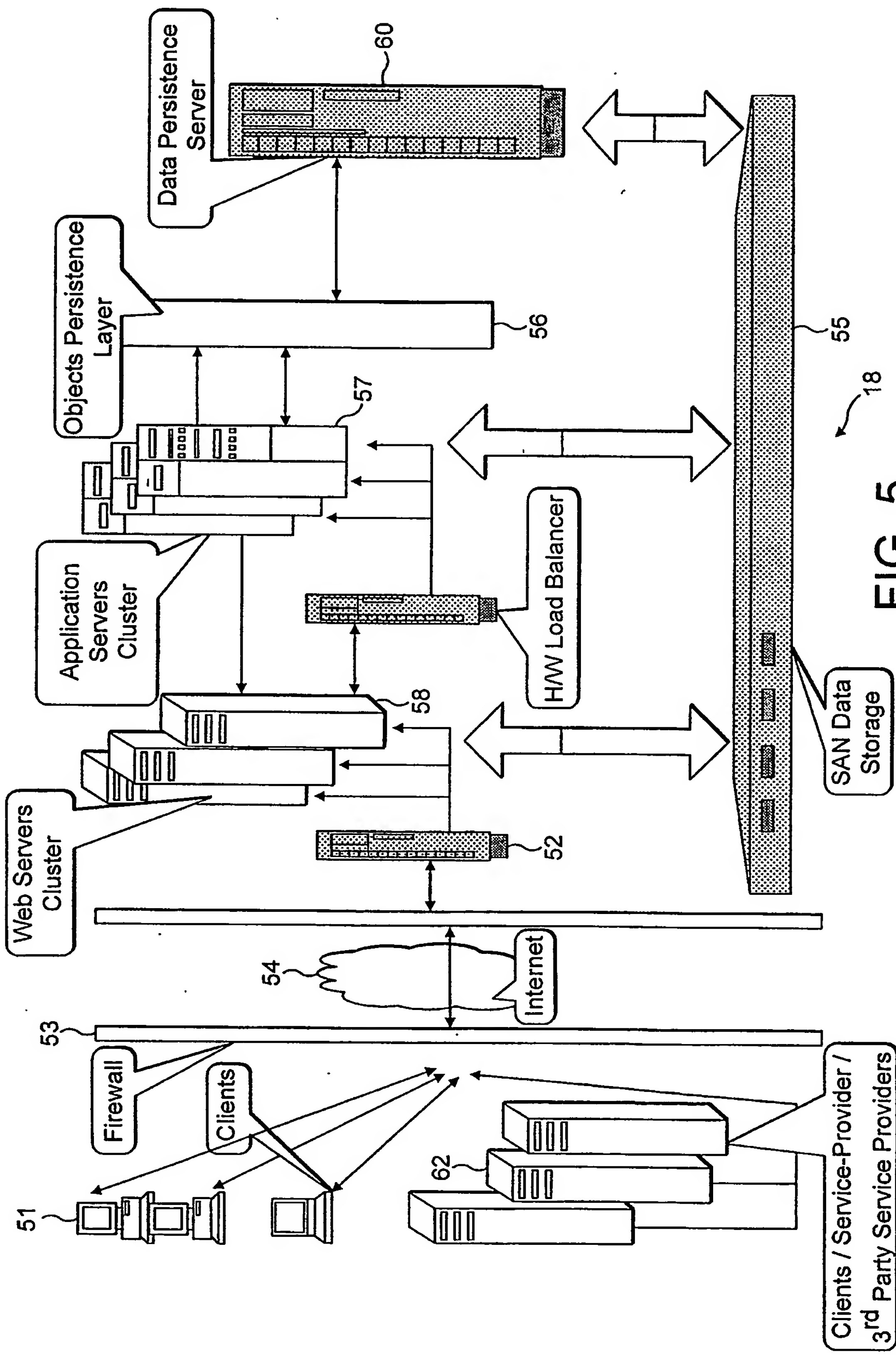

Data Persistence Server
60
Objects Persistence Layer
56
57
Application Servers Cluster
H/W Load Balancer
58
Web Servers Cluster
52
55
18
SAN Data Storage
Internet
54
53
Firewall
Clients
51
62
Clients / Service-Provider / 3rd Party Service Providers

FIG. 5

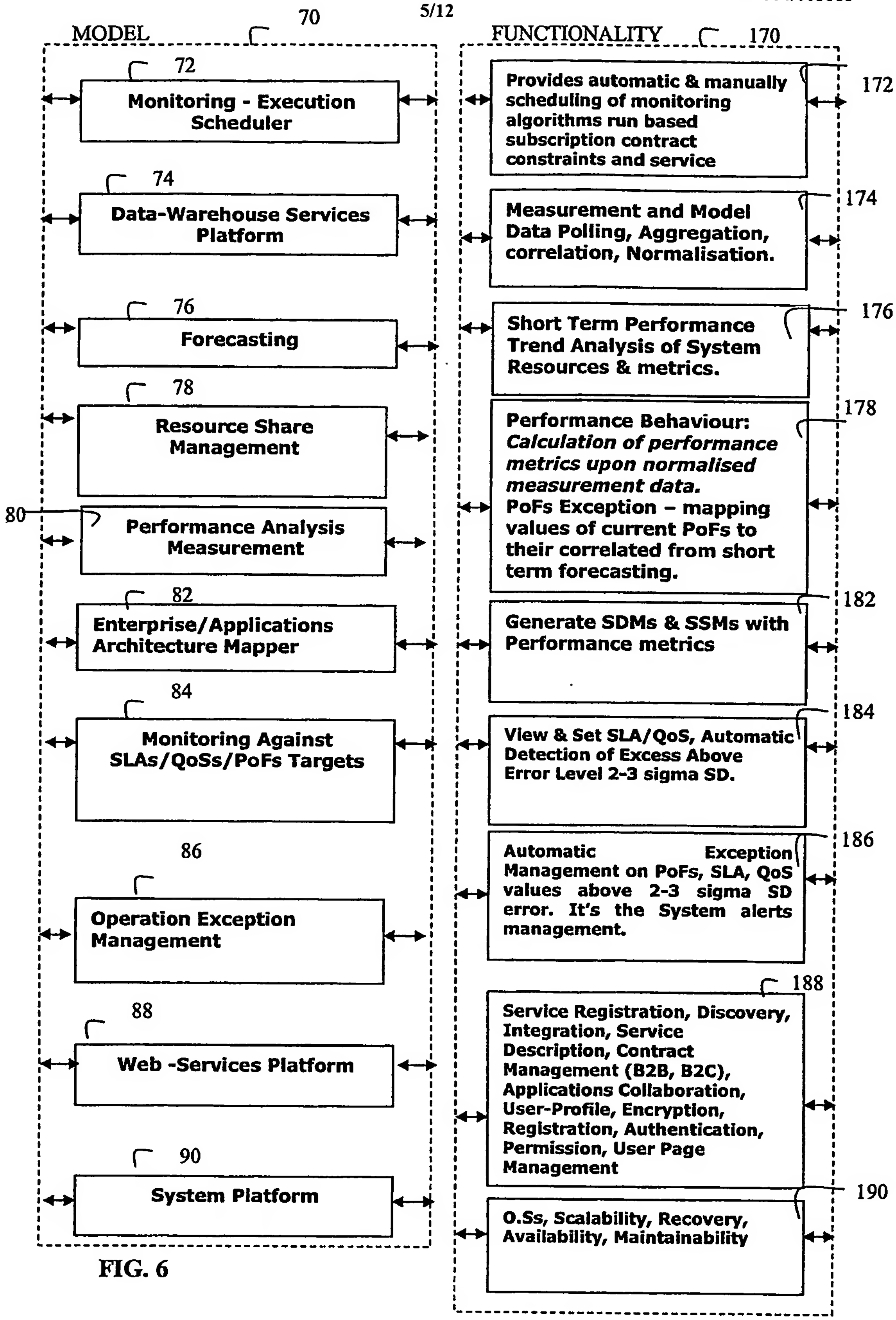

MODEL
70
72
Monitoring - Execution Scheduler
74
Data-Warehouse Services Platform
76
Forecasting
78
Resource Share Management
80
Performance Analysis Measurement
82
Enterprise/Applications Architecture Mapper
84
Monitoring Against SLAs/QoSs/PoFs Targets
86
Operation Exception Management
88
Web -Services Platform
90
System Platform
FUNCTIONALITY
170
172
Provides automatic & manually scheduling of monitoring algorithms run based subscription contract constraints and service
174
Measurement and Model Data Polling, Aggregation, correlation, Normalisation.
176
Short Term Performance Trend Analysis of System Resources & metrics.
178
Performance Behaviour: Calculation of performance metrics upon normalised measurement data. PoFs Exception – mapping values of current PoFs to their correlated from short term forecasting.
182
Generate SDMs & SSMs with Performance metrics
184
View & Set SLA/QoS, Automatic Detection of Excess Above Error Level 2-3 sigma SD.
186
Automatic Exception Management on PoFs, SLA, QoS values above 2-3 sigma SD error. It's the System alerts management.
188
Service Registration, Discovery, Integration, Service Description, Contract Management (B2B, B2C), Applications Collaboration, User-Profile, Encryption, Registration, Authentication, Permission, User Page Management
190
O.Ss, Scalability, Recovery, Availability, Maintainability

FIG. 6

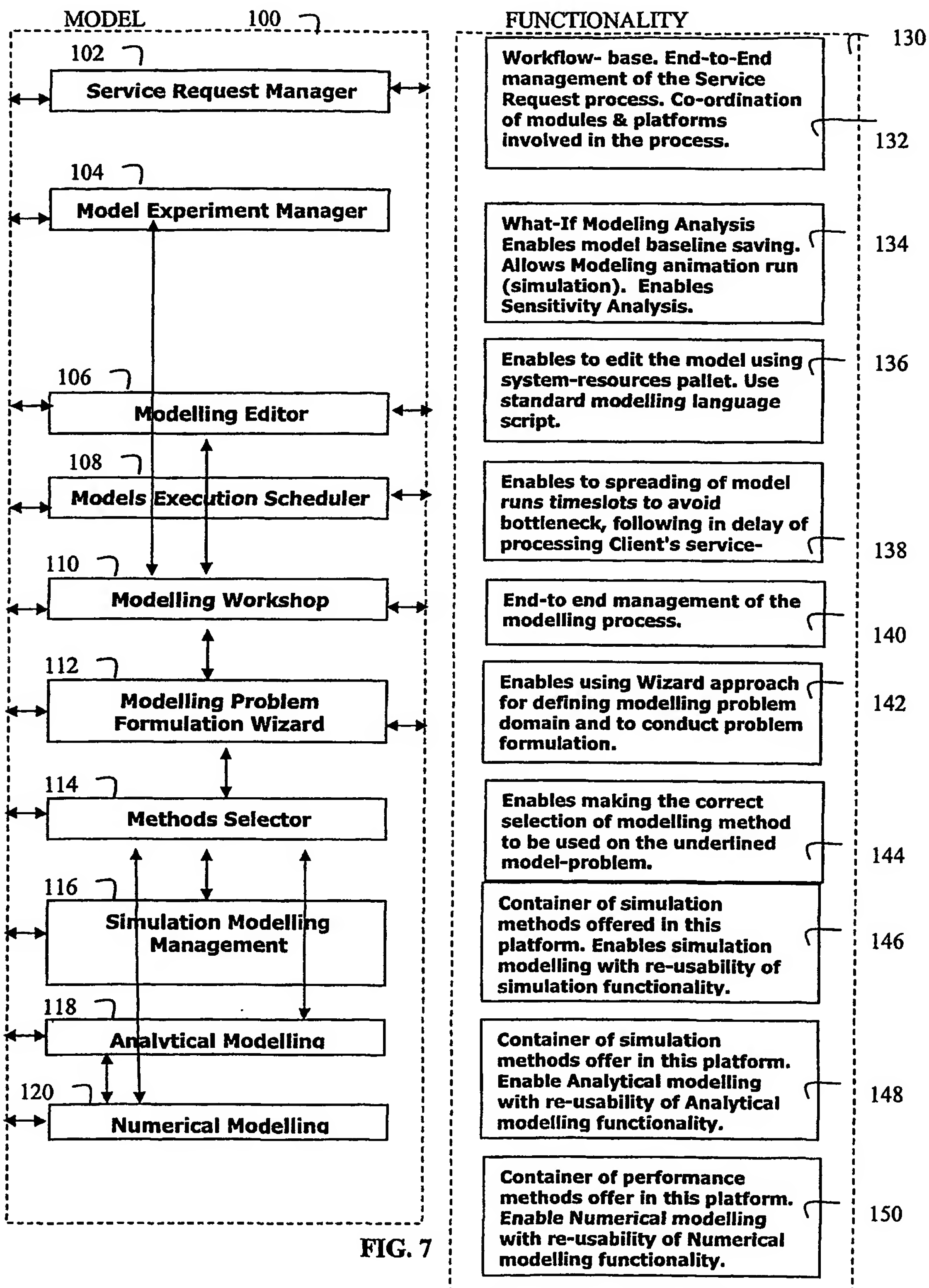
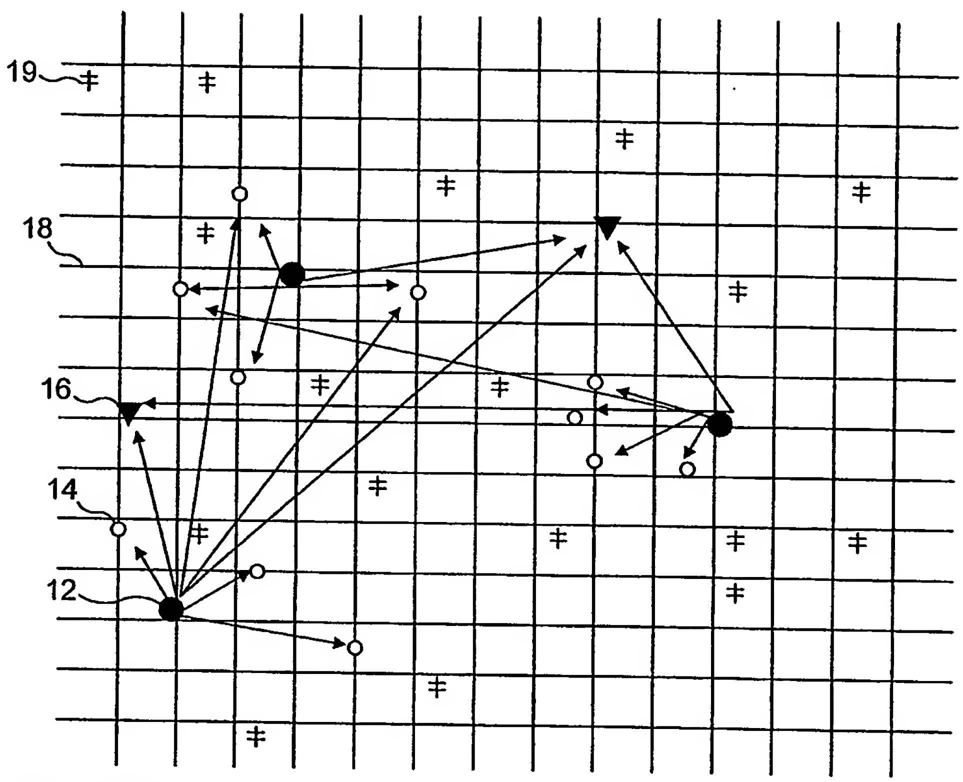

MODEL
100
FUNCTIONALITY
130
102
Service Request Manager
104
Model Experiment Manager
106
Modelling Editor
108
Models Execution Scheduler
110
Modelling Workshop
112
Modelling Problem Formulation Wizard
114
Methods Selector
116
Simulation Modelling Management
118
Analytical Modelling
120
Numerical Modelling
Workflow- base. End-to-End management of the Service Request process. Co-ordination of modules & platforms involved in the process.
132
What-If Modeling Analysis Enables model baseline saving. Allows Modeling animation run (simulation). Enables Sensitivity Analysis.
134
Enables to edit the model using system-resources pallet. Use standard modelling language script.
136
Enables to spreading of model runs timeslots to avoid bottleneck, following in delay of processing Client's service-
138
End-to end management of the modelling process.
140
Enables using Wizard approach for defining modelling problem domain and to conduct problem formulation.
142
Enables making the correct selection of modelling method to be used on the underlined model-problem.
144
Container of simulation methods offered in this platform. Enables simulation modelling with re-usability of simulation functionality.
146
Container of simulation methods offer in this platform. Enable Analytical modelling with re-usability of Analytical modelling functionality.
148
Container of performance methods offer in this platform. Enable Numerical modelling with re-usability of Numerical modelling functionality.
150

FIG. 7



Map Index :

- Computing Centre sign: ●
- Service-Providers Local Office Analysts sign: o
- Service-Providers Background Analysts sign: ▼
- Collaboration System : Grid lines
- 3rd Party Service Providers sign: ‡

FIG. 8

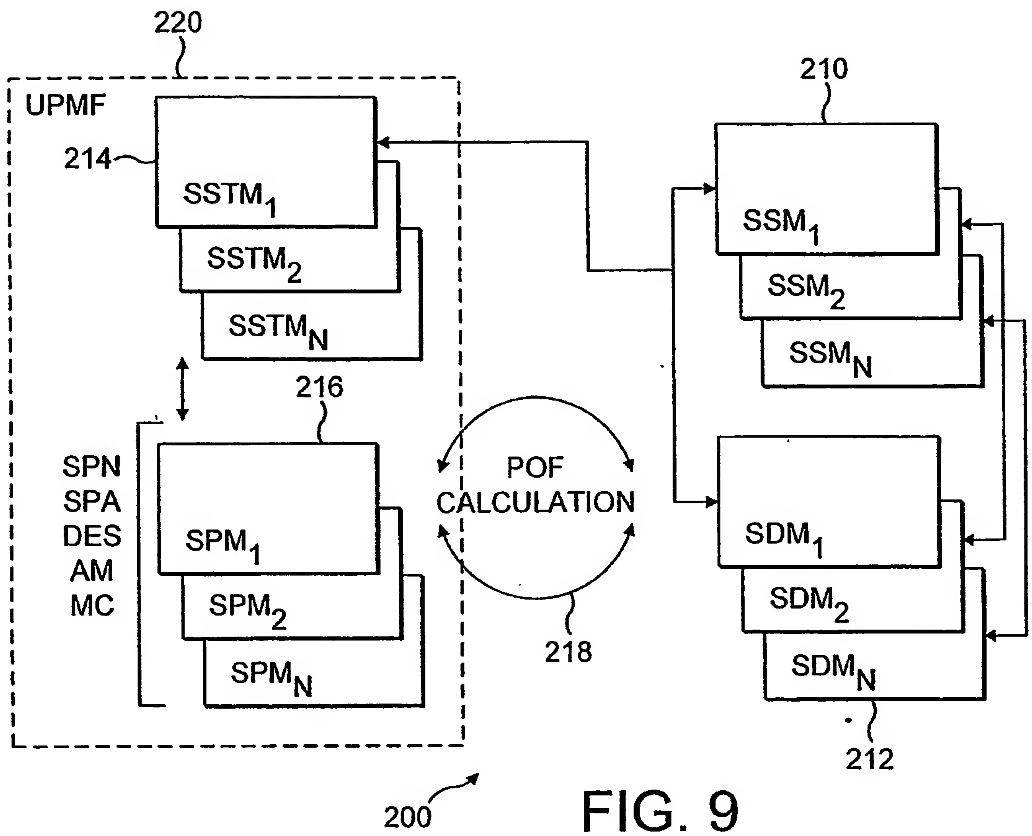

220
UPMF
214
SSTM1
SSTM2
SSTMN
216
SPN
SPA
DES
AM
MC
SPM1
SPM2
SPMN
POF CALCULATION
218
210
SSM1
SSM2
SSMN
SDM1
SDM2
SDMN
212
200

FIG. 9

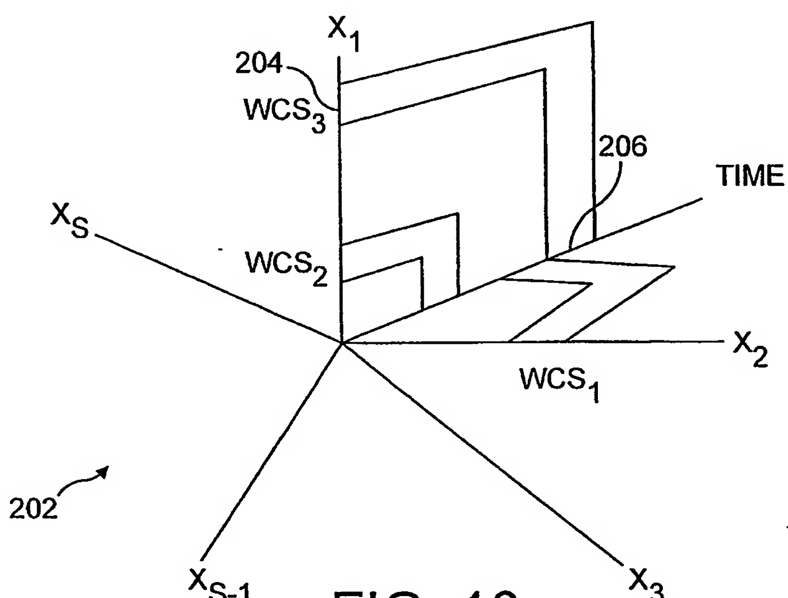

X1
204
WCS3
206
TIME
XS
WCS2
X2
WCS1
202
XS-1
X3

FIG. 10

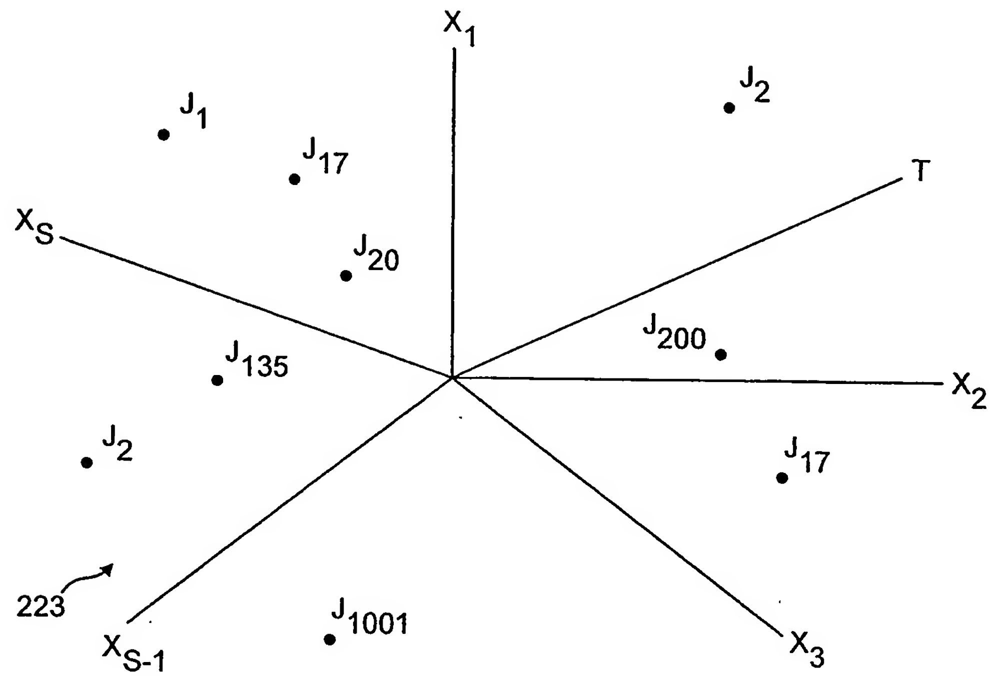


FIG. 11A

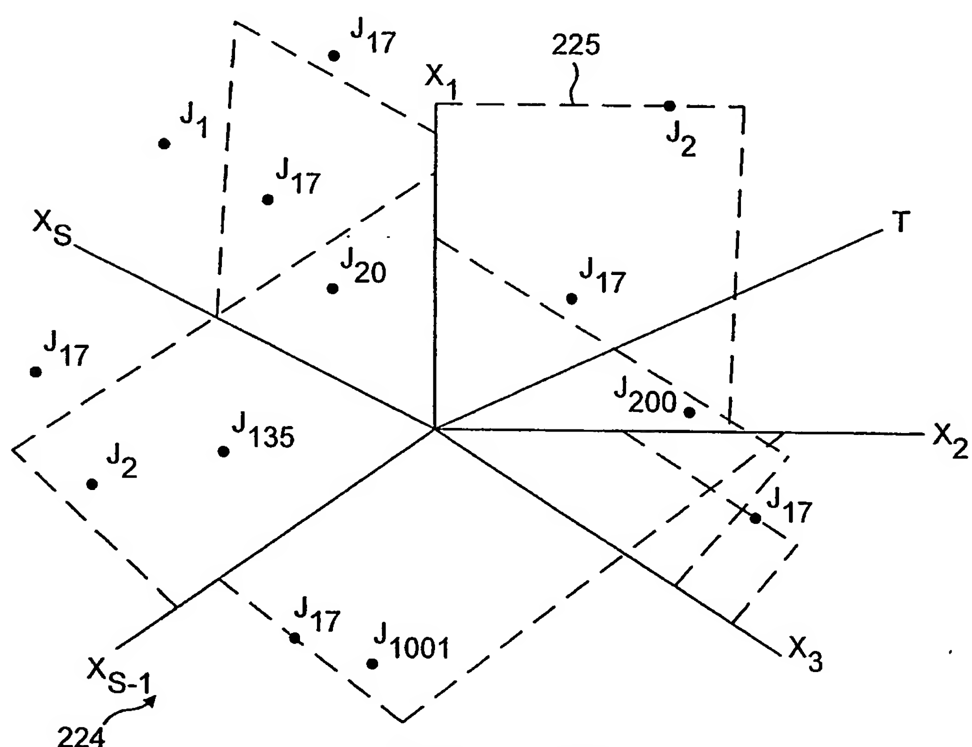


FIG. 11B

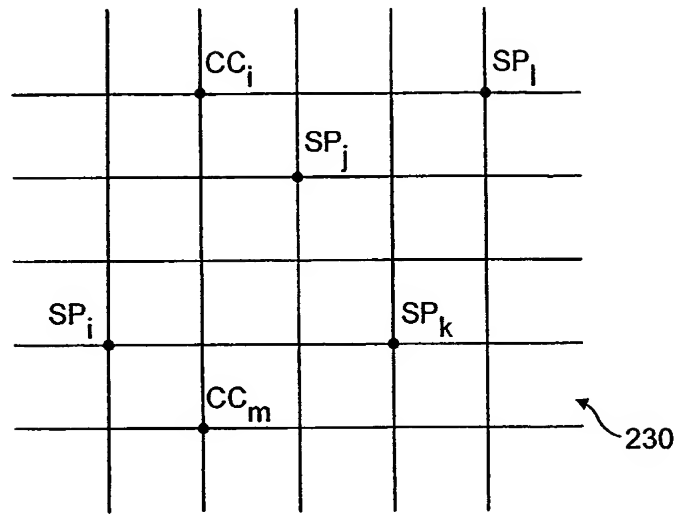


FIG. 12

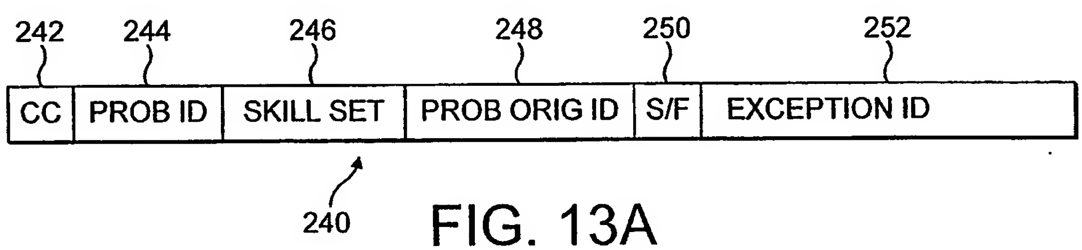


FIG. 13A

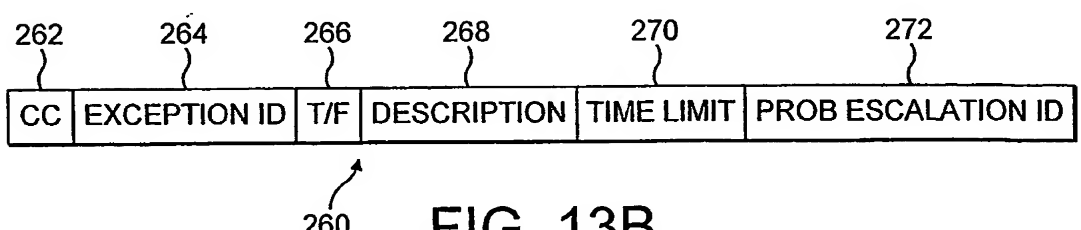


FIG. 13B

Silver Subscription			
Total Consulting (support for 3 application, Excluding Initial Const)	*per year $26,895*	*per year £ 14,538*	*pm $2,242*
	S/W	*S/W & Consultancy*	*pm (All)*
Target Systems — *2 X Workgroup Tier + 2 X Department Tier + 3 X Application + 3 X Universal.D + 3 X Switches & Routers*	*$21,070*	*$47,965*	*$3,998*
GBP	*£ 11,389*	*£ 25,927*	*£2,161*
Gold Subscription			
(weekly report basis) *Total Consulting (support for 3 application)*	*per year $79,283*	*per year £ 42,856*	*pm $6,607*
	S/W	*S/W & Consultancy*	*pm (All)*
Target Systems — *2 X Workgroup Tier + 2 X Department Tier + 3 X Application + 3 X Universal.D + 3 X Switches & Routers*	*$21,070*	*$100,353*	*$8,363*
GBP	*£ 11,389*	*£ 54,245*	*£4,521*
Platinum Subscription			
(weekly report basis, plus ad-hoc reporting) *Total Consulting*	*per year $146,323*	*per year £ 79,093*	*pm $12,194*
	S/W	*S/W & Consultancy*	*pm (All)*
Target Systems — *2 X Workgroup Tier + 2 X Department Tier + 3 X Application + 3 X Universal.D + 3 X Switches & Routers*	*$21,070*	*$167,393*	*$13,950*
GBP	*£ 11,389*	*£ 90,482*	*£7,541*

285

FIG. 14

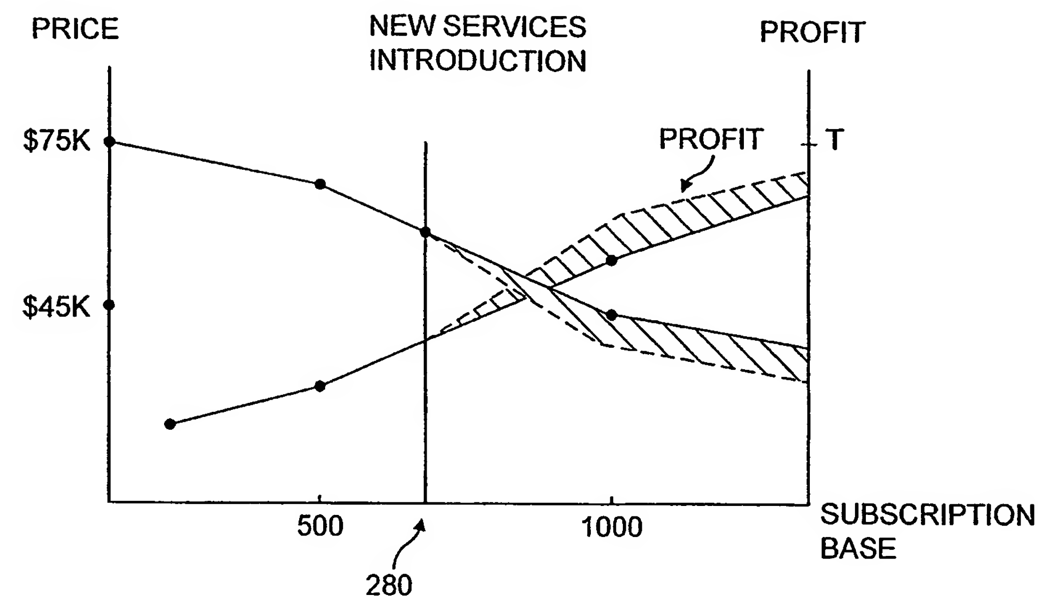

PRICE
NEW SERVICES INTRODUCTION
PROFIT
$75K
$45K
PROFIT
T
500
1000
SUBSCRIPTION BASE
280

FIG. 15

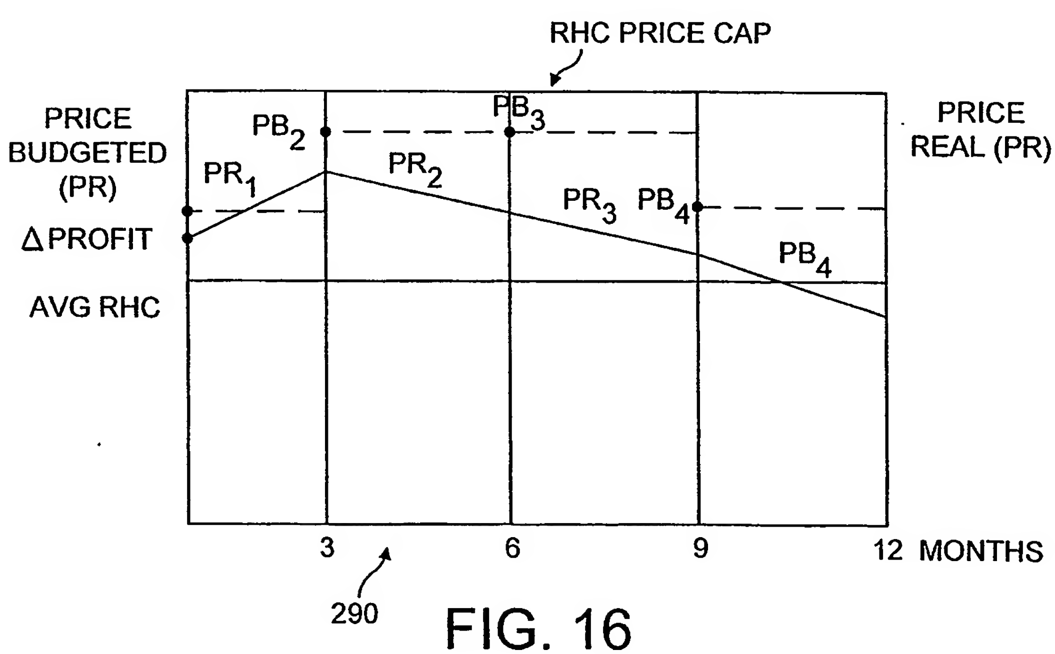

RHC PRICE CAP
PRICE BUDGETED (PR)
ΔPROFIT
AVG RHC
PRICE REAL (PR)
PB2
PB3
PR1
PR2
PR3
PB4
PB4
3
6
9
12 MONTHS
290

FIG. 16